

AD-A064 896

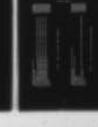
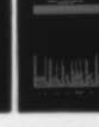
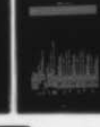
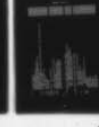
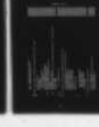
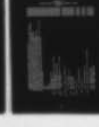
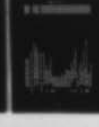
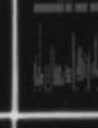
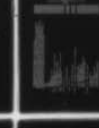
NAVAL POSTGRADUATE SCHOOL MONTEREY CALIF
REAL TIME KALMAN FILTERING FOR TORPEDO RANGE TRACKING.(U)
DEC 78 D M DWYER

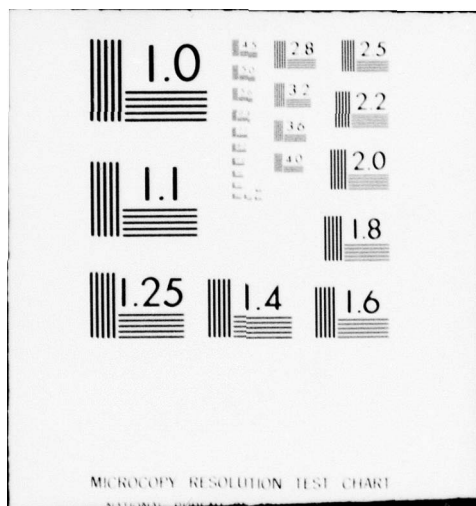
F/6 17/1

UNCLASSIFIED

NL

1 OF 2
AD
A 064 896





LEVEL II

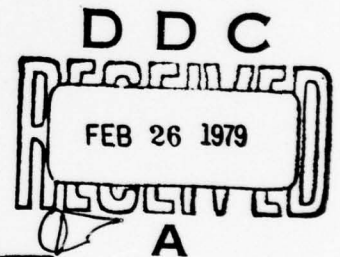
2
nu

ADA064896

NAVAL POSTGRADUATE SCHOOL
Monterey, California



DDC FILE COPY



THESIS

REAL TIME KALMAN FILTERING FOR
TORPEDO RANGE TRACKING

by

Dennis Michael Dwyer

December 1978

Thesis Advisor:

H. A. Titus

Approved for public release; distribution unlimited

79 02 23 129

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

| REPORT DOCUMENTATION PAGE | | READ INSTRUCTIONS BEFORE COMPLETING FORM |
|---|-----------------------|--|
| 1. REPORT NUMBER | 2. GOVT ACCESSION NO. | 3. RECIPIENT'S CATALOG NUMBER |
| 4. TITLE (and Subtitle) REAL TIME KALMAN FILTERING FOR TORPEDO RANGE TRACKING | | 5. TYPE OF REPORT & PERIOD COVERED Master's Thesis December 1978 |
| 7. AUTHOR(s) Dennis Michael Dwyer | | 6. PERFORMING ORG. REPORT NUMBER |
| 9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940 | | 8. CONTRACT OR GRANT NUMBER(s) |
| 11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940 | | 10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 12 99P |
| 14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) | | 12. REPORT DATE December 1978 |
| | | 13. NUMBER OF PAGES 98 |
| | | 15. SECURITY CLASS. (of this report) Unclassified |
| | | 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE |
| 16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited | | |
| 17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report) | | |
| 18. SUPPLEMENTARY NOTES | | |
| 19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Torpedo Tracking Kalman Filter | | |
| 20. ABSTRACT (Continue on reverse side if necessary and identify by block number) Two Extended Kalman filter routines, one using a one-step estimation/prediction and the other a sequential approach, were developed and compared to provide real time estimates of target positions on the three dimensional underwater tracking range at Naval Underwater Weapons Engineering Station, Keyport, Washington. Inputs to the routines were acoustic pulse transit times from the target to receiving array elements which are non-linear functions of the position coordinates. These inputs were linearized and the | | |

DD FORM 1473
1 JAN 73
(Page 1)EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

1

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

251 450

JLC

(cont)
UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE/When Data Entered

→ filter gains calculated on-line. Simulated runs were conducted for tracks in the area of one hydrophone array and for tracks that transited through multiple arrays. It was found that the sequential estimate routine exhibited better performance in recovering from transients caused by random measurement noise or target movement.

| | |
|---------------------------------|---|
| ACCESSION NO. | |
| NTIS | White Section <input checked="" type="checkbox"/> |
| FOUO | Diff Section <input type="checkbox"/> |
| ORIGINATOR | <input type="checkbox"/> |
| JUSTIFICATION | |
| BY | |
| DISTRIBUTION/AVAILABILITY CODES | |
| SQL | AVAIL. AND IN SPECIAL |
| A | |

79 02 23 129

Approved for public release; distribution unlimited

REAL TIME KALMAN FILTERING

FOR

TORPEDO RANGE TRACKING

by

Dennis Michael Dwyer
Lieutenant, United States Navy
B.S., United States Naval Academy, 1973

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
December 1978

Author

DM Dwyer

Approved by:

A/S Teters

Thesis Advisor

Geo L Jackson

Second Reader

WZ Kirk

Chairman, Department of Electrical Engineering

William M. Folles

Dean of Science and Engineering

ABSTRACT

Two Extended Kalman filter routines, one using a one-step estimation/prediction and the other a sequential approach, were developed and compared to provide real time estimates of target positions on the three dimensional underwater tracking range at Naval Underwater Weapons Engineering Station, Keyport, Washington. Inputs to the routines were acoustic pulse transit times from the target to receiving array elements which are non-linear functions of the position coordinates. These inputs were linearized and the filter gains calculated on-line. Simulated runs were conducted for tracks in the area of one hydrophone array and for tracks that transited through multiple arrays. It was found that the sequential estimate routine exhibited better performance in recovering from transients caused by random measurement noise or target movement.

TABLE OF CONTENTS

| | | |
|-------------|--|----|
| I. | INTRODUCTION - - - - - | 7 |
| II. | THREE DIMENSIONAL RANGE DESCRIPTION- - - - - | 9 |
| III. | THEORY - - - - - | 12 |
| | A. EXTENDED KALMAN FILTER - - - - - | 12 |
| | B. SEQUENTIAL EXTENDED KALMAN FILTER- - - - - | 14 |
| IV. | PROBLEM DEFINITION - TORPEDO TRACKING WITH THE EXTENDED KALMAN FILTER - - - - - | 16 |
| | A. THE SEQUENTIAL APPROACH- - - - - | 17 |
| | B. THE TRADITIONAL MATRIX INVERSION APPROACH- - | 21 |
| V. | TESTING AND SIMULATION - - - - - | 22 |
| | A. THE GATING SCHEME- - - - - | 23 |
| | B. MULTIPLE ARRAY TRACKING- - - - - | 25 |
| VI. | SIMULATION RESULTS - - - - - | 28 |
| VII. | CONCLUSIONS- - - - - | 70 |
| APPENDIX A: | PROGRAM DESCRIPTION AND FEATURES- - - - - | 73 |
| | A. TIME VARYING STATE TRANSITION MATRIX- - | 76 |
| | B. ADAPTIVE Q- - - - - | 78 |
| APPENDIX B: | SEQUENTIAL EXTENDED KALMAN FILTER PROGRAM LISTING - - - - - | 80 |
| APPENDIX C: | TRADITIONAL EXTENDED KALMAN FILTER PROGRAM LISTING - - - - - | 85 |
| APPENDIX D: | UTILITY PROGRAM LISTING - - - - - | 91 |
| | LIST OF REFERENCES - - - - - | 97 |
| | INITIAL DISTRIBUTION LIST- - - - - | 98 |

ACKNOWLEDGEMENT

The author is deeply indebted to Professor Hal Titus for his counsel during this project and also to Lieutenant Commander John Gauss for his professional guidance and assistance.

I. INTRODUCTION

The Naval Underwater Weapons Engineering Station, Keyport, Washington currently operates two three-dimensional underwater tracking ranges with the capability of acoustically tracking torpedoes. Underwater tracking employs an acoustic device installed in the object to be tracked. This device transmits timed acoustic pulses which are received by bottom mounted hydrophone arrays and then relayed via cable to a computer at the observation site which calculates the position of the object for each pulse and plots its path.

The measured data, which is the time elapsed from transmission of a pulse until its receipt at the hydrophone array, is corrupted by noise due to the combined effects of environmental factors and the measurement instruments.

These noisy tracks are later analyzed, and measurements judged most inaccurate on the basis of total track statistics are removed in order to obtain a smooth representation of the track.

As stated in Reference 1, the computer system at the Dabob range of the station will be up-graded, and will consist of three MODCOMP IV computers by DATACOM, Inc. A software conversion project will take place with applications software being developed for the new computers. The bulk of this development will consist of converting the current tracking programs and other related programs to FORTRAN.

An opportunity exists for expanding the real-time capability of the system by applying a Kalman filter routine which can take as an input the transit times of the acoustic pulses, and produce the best estimate of the position of the tracked object at a particular time.

II. THREE DIMENSIONAL RANGE DESCRIPTION

The three dimensional range described in Reference 2 is an acoustic system capable of determining the trajectory of suitably instrumented underwater objects in the vicinity of a transducer array placed at the bottom of the bay. The tracked unit (torpedo) carries a synchronous clock and an acoustic transducer. A hydrophone array defines a rectangular coordinate system to which measurements are referred. Positional information is obtained from the transit times of a periodic pulsed acoustic signal traveling from the torpedo to four independent hydrophones located on each array. The geometry of the hydrophones and the coordinate system is illustrated in Figure 1. On each array, the four hydrophones R_C , R_X , R_Y and R_Z are on four adjacent vertices of a cube. Each hydrophone is separated by a distance $d = 30$ feet along the edges. The origin of the coordinate system is at the center of the cube.

The transit times of the acoustic pulse from the tracked object to each of the four independent hydrophones can be expressed as follows:

$$T_C = 1/VEL \left[(X+d/2)^2 + (Y+d/2)^2 + (Z+d/2)^2 \right]^{1/2}$$

$$T_X = 1/VEL \left[(X-d/2)^2 + (Y+d/2)^2 + (Z+d/2)^2 \right]^{1/2}$$

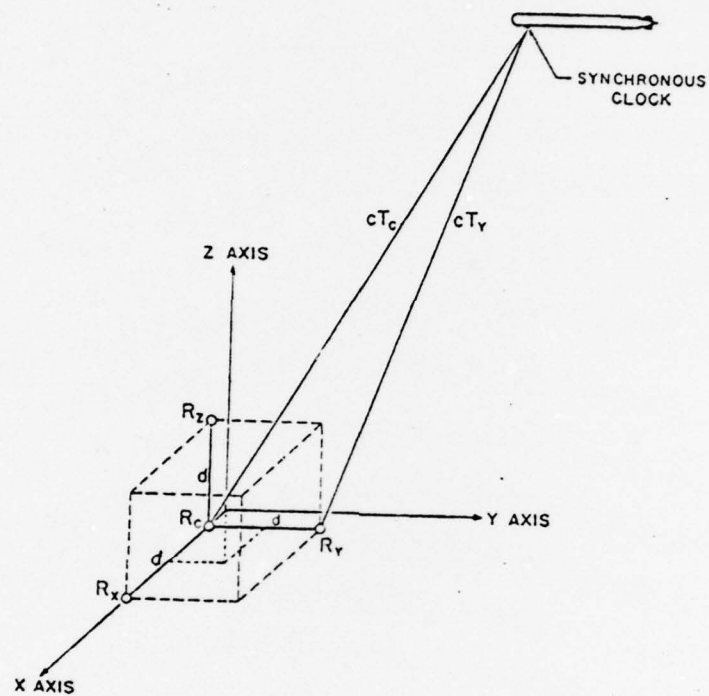


FIGURE 1: The Geometry Used in the Calculation of Positional Coordinates (The origin is at the center of the dashed cube).

$$T_Y = 1/VEL \left[(X+d/2)^2 + (Y-d/2)^2 + (Z+d/2)^2 \right]^{1/2}$$

$$T_Z = 1/VEL \left[(X+d/2)^2 + (Y+d/2)^2 + (Z-d/2)^2 \right]^{1/2}$$

Where VEL equals the velocity of propagation of sound in water and C, X, Y and Z are the four hydrophones on each array. It is essential as part of this fundamental calculation to know when the tracked object emits an acoustic pulse in order to measure the transit times to the hydrophones. For submerged objects two stable crystal-controlled clocks, one in the tracked unit and the other at the computer are used. Prior to a run the two clocks are synchronized by radio.

The range is a high frequency, short-baseline facility with the acoustic tracking pulses emitted at 75 kHz. The range layout consists of six in-line hydrophone arrays spaced 2000 yards apart.

III. THEORY

In Reference 3, a Kalman filter application was used assuming a linear system and filtering on the corrupted X, Y and Z positions that the computer had already calculated from the received transit times of the acoustic signals.

A. THE EXTENDED KALMAN FILTER

Since the transit times were readily available and are non-linear functions of position, these equations can be linearized and Kalman filter theory applied using the extended Kalman filter. This procedure produces a real-time system, filtering on the corrupted transit times T_C , T_X , T_Y and T_Z , without the necessity of converting these times to positions.

For tracking, a fifth order state vector was chosen:

$$\tilde{X} = \begin{bmatrix} X \\ \dot{X} \\ X \\ Y \\ \dot{Y} \\ Y \\ Z \end{bmatrix}$$

The target was assumed to maintain constant depth and any velocity in the Z direction (\dot{Z}) was considered a random excitation.

The states were characterized by the following difference equation:

$$\tilde{X}(K+1) = \Phi \tilde{X}(K) + \Gamma \tilde{W}(K)$$

and the noisy measurement equation:

$$\tilde{z}(K) = M(\tilde{x}) + \tilde{v}(K)$$

where

$\tilde{x}(K)$ is the N-dimensional state vector at time K

$\tilde{w}(K)$ is the M-dimensional random forcing input at time K

$\tilde{z}(K)$ is the J-dimensional measurement vector at time K

$\tilde{v}(K)$ is the J-dimensional random noise vector at time K and noise is assumed white and zero-mean Gaussian

Φ and Γ are constant matrices

$M(\tilde{x})$ is a matrix of the non-linear measurement equations which are a function of the states.

The estimator equations are given by:

$$\hat{\tilde{x}}(K/K) = \hat{\tilde{x}}(K/K-1) + G(K) * \left[\tilde{z}(K) - M(\tilde{x}) * \hat{\tilde{x}}(K/K-1) \right]$$

and

$$\tilde{P}(K/K) = \left[I - G(K) \tilde{H}(K) \right] \tilde{P}(K/K-1)$$

where

$\hat{\tilde{x}}(K/K)$ is the estimate of the state at time K given K measurements

$\tilde{P}(K/K)$ is the covariance of estimation error matrix at time K

$G(K)$ is the Kalman filter gain at time K

which is defined as:

$$G(K) = \tilde{P}(K/K-1) \tilde{H}(K)^T \left[\tilde{H}(K) \tilde{P}(K/K-1) \tilde{H}(K)^T + \tilde{R}(K) \right]^{-1}$$

$\tilde{R}(K)$ is the covariance of random measurement noise matrix

$\tilde{H}(K)$ is the matrix used to linearize the non-linear measurement equations. It is defined as:

$$H(K) = \left. \frac{\partial M(\tilde{x})}{\partial \tilde{x}} \right|_{\hat{\tilde{x}}(K/K-1)}$$

The measurement equation matrix $M(\tilde{x})$ is expanded in a Taylor series and linearized around the one step prediction based on the last estimate. Only first order terms are kept.

The prediction equations are given by:

$$\hat{\tilde{x}}(K+1/K) = \Phi \hat{\tilde{x}}(K/K) + \Gamma W(K)$$

$$P(K+1/K) = \Phi P(K/K) \Phi^T + Q(K)$$

where

$Q(K)$ is the covariance of random excitation matrix found by:

$$Q(K) = \Gamma \text{COV}(W) \Gamma^T$$

and $\text{COV}(W)$ is the covariance of the random forcing input or acceleration.

B. THE SEQUENTIAL EXTENDED KALMAN FILTER

The prediction and estimation equations just mentioned represent a system that is characterized by a one-step estimation and prediction for each set of new measurements. Some important points in this one step process must be stressed:

1. New estimates are only available after the prediction and estimation equations are calculated.
2. A major part of the time period for the estimation calculation is spent in the gain equation because of the inversion of the (JXJ) matrix that is required, where J is the number of observations.

3. The one-step prediction and estimation for the extended Kalman filter is based on a linearization about the predicted value over a period t_K to t_{K+1} which might be extended.

When the measurements come from statistically independent sources, like the hydrophones in the torpedo tracking problem, a sequential approach can be taken to process each arrival time separately.

If the measurements are assumed to occur simultaneously, they can be processed one at a time and the result of processing one measurement component is used in the following computation to process the next measurement component.

IV. PROBLEM DEFINITION - TORPEDO TRACKING WITH THE EXTENDED KALMAN FILTER

In the torpedo tracking problem, the non-linear observations are the four independent transit times from the tracked object to the hydrophones, T_C , T_X , T_Y and T_Z . Thus the non-linear measurement matrix $M(\tilde{X})$ is defined as:

$$M(\tilde{X}) = \begin{bmatrix} T_C \\ T_X \\ T_Y \\ T_Z \end{bmatrix} = \begin{bmatrix} \frac{1}{VEL} \left[(X+d/2)^2 + (Y+d/2)^2 + (Z+d/2)^2 \right]^{1/2} \\ \frac{1}{VEL} \left[(X-d/2)^2 + (Y+d/2)^2 + (Z+d/2)^2 \right]^{1/2} \\ \frac{1}{VEL} \left[(X+d/2)^2 + (Y-d/2)^2 + (Z+d/2)^2 \right]^{1/2} \\ \frac{1}{VEL} \left[(X+d/2)^2 + (Y+d/2)^2 + (Z-d/2)^2 \right]^{1/2} \end{bmatrix}$$

$M(\tilde{X})$ is expanded into a Taylor series and linearly approximated by:

$$M(\tilde{X}) \approx \left. \frac{\partial M(\tilde{X})}{\partial \tilde{X}} \right|_{\hat{\tilde{X}}(K/K-1)} \tilde{X} = H(\tilde{X}) \left. \tilde{X} \right|_{\hat{\tilde{X}}(K/K-1)}$$

The linearizing H matrix is evaluated around the best information available at the time which is the prediction $\hat{\tilde{X}}(K/K-1)$, and is used in the calculation of the gain $\tilde{G}(K)$ and estimated covariance of error $\tilde{P}(K/K)$ equations.

The torpedo dynamics used for the tracking problem are assumed to be $1/S^2$ with extimations on five states X position,

X velocity, Y position, Y velocity and Z position (height of torpedo above hydrophone array). The mean of the random excitation is assumed to be zero $\left[E(W(K))=0\right]$, and this simplifies the estimate prediction equation to:

$$\hat{\tilde{X}}(K+1/K) = \phi \hat{\tilde{X}}(K/K)$$

Also the mean of the random noise is assumed to be zero $\left[E(V(K))=0\right]$. In forming the measurement equation, the best guess of $v(K)$ is used which is the mean producing:

$$\tilde{Z}(K) = M(\tilde{X})$$

Four measurements are taken every 1.31 seconds, which is one time slot, and with this sampling time the $1/s^2$ plant has state transition (PHI) and gamma matrices equal to:

$$\phi = \begin{bmatrix} 1 & 1.31 & 0 & 0 & 0 \\ 0 & 1. & 0 & 0 & 0 \\ 0 & 0 & 1. & 1.31 & 0 \\ 0 & 0 & 0 & 1. & 0 \\ 0 & 0 & 0 & 0 & 1. \end{bmatrix}$$

and

$$\Gamma = \begin{bmatrix} .86 & 0 & 0 \\ 1.31 & 0 & 0 \\ 0 & .86 & 0 \\ 0 & 1.31 & 0 \\ 0 & 0 & 1.31 \end{bmatrix}$$

A. THE SEQUENTIAL APPROACH

In the sequential approach, the basic Kalman filter equations

have been modified to circumvent the matrix inversion in the gain equation and to obtain a more accurate estimate. Calculations are performed on each of the four independent transit times in the following order T_C , T_X , T_Y and T_Z for each 1.31 second time slot.

The estimate of the states, $\hat{\tilde{X}}(K/K)$, based on one time measurement is used as the prediction $\hat{\tilde{X}}(K/K-1)$ for the calculations on the next measurement.

In this manner only parts of the linearizing H matrix and gain matrices are used in each calculation.

After the linearizing H matrix is formed from

$$\tilde{H}(\tilde{X}) = \left. \frac{\partial M(\tilde{X})}{\partial \tilde{X}} \right|_{\hat{\tilde{X}}(K/K-1)}$$

evaluated around the initial states $\hat{\tilde{X}}(1/0)$ for $K=1$, the first gain column corresponding to the first time measurement T_C is calculated from:

$$\tilde{G}_{iCOL} = \frac{\tilde{P}(K/K-1) \tilde{H}_{iROW}^T}{\tilde{H}_{iROW} \tilde{P}(K/K-1) \tilde{H}_{iROW}^T + R_{ii}}$$

where $i = 1$ to J , and J is the number of observations. In the tracking problem $J = 4$ corresponding to the four measured times.

Thus, the first row of the H matrix is used to calculate the first column of the gain matrix with both corresponding to the first measured time T_C .

Next, an estimate of the particular observation time $\hat{M}(X)$ is calculated

$$\hat{M}(X) = \begin{bmatrix} \hat{T}_C \\ \hat{T}_X \\ \hat{T}_Y \\ \hat{T}_Z \end{bmatrix} = \begin{bmatrix} \frac{1}{VEL} \left[(X+d/2)^2 + (Y+d/2)^2 + (Z+d/2)^2 \right] & \frac{1}{2} \\ \frac{1}{VEL} \left[(X-d/2)^2 + (Y+d/2)^2 + (Z+d/2)^2 \right] & \frac{1}{2} \\ \frac{1}{VEL} \left[(X+d/2)^2 + (Y-d/2)^2 + (Z+d/2)^2 \right] & \frac{1}{2} \\ \frac{1}{VEL} \left[(X+d/2)^2 + (Y+d/2)^2 + (Z-d/2)^2 \right] & \frac{1}{2} \end{bmatrix}$$

Using the predicted values of X, Y, Z from $\hat{X}(K/K-1)$.

The difference between the observed transit time Z_i and the estimated transit time \hat{T}_i forms the residual Z_{DIFF} which is used in the estimate equation.

$$\hat{X}_i = \hat{X}(K/K-1) + G_i \text{COL} \begin{bmatrix} Z_{DIFF} \end{bmatrix}$$

This equation gives an estimate of the states based on one measurement.

Next, the covariance of estimation error is calculated based on one measurement using:

$$P_i = \begin{bmatrix} I \\ \sim \end{bmatrix} - G_i \text{COL} H_i \text{ROW} \quad P_{i-1}$$

where

\tilde{I} equals the identity matrix

P_{i-1} is the theoretical covariance of estimation error from the previous measurement or if $i=1$, the prediction $\tilde{P}(K/K-1)$

After the first iteration, \tilde{X}_1 becomes $\tilde{X}(K/K-1)$ and P_1 becomes $\tilde{P}(K/K-1)$ for the second iteration which calculates the estimate of the states based on the second measurement T_X .

After four iterations ($i=4$), \tilde{X}_4 becomes the estimate for the time slot, $\hat{X}(K/K)$ and P_4 becomes the updated covariance of error $\tilde{P}(K/K)$.

Then the predictions for the next time slot are calculated using:

$$\tilde{X}(K+1/K) = \phi \tilde{X}(K/K)$$

and

$$\tilde{P}(K+1/K) = \phi \tilde{P}(K/K) \phi^T + Q(K)$$

The entire process is repeated for the next set of measurements forming a sequential, extended Kalman filter to produce real time estimates of the torpedo track.

A listing of the FORTRAN program designed for the sequential extended Kalman filter is contained in Appendix B. The program is in modular form and well documented by comments for ease of implementation. All repetitive calculations and utility routines are separated into subroutines and are listed in Appendix D.

B. THE MATRIX INVERSION APPROACH

All results obtained using the sequential, extended Kalman filter were compared to results from a traditional extended Kalman filter using the equations delineated in the THEORY section. This includes an inversion of a 4x4 matrix in the gain equation.

$$\tilde{G}(K) = \tilde{P}(K/K-1)\tilde{H}(K)^T \left[\tilde{H}(K)\tilde{P}(K/K-1)\tilde{H}(K)^T + \tilde{R}(K) \right]^{-1}$$

The portion of the equation to be inverted is always symmetric and the IBM-360 library subroutine SINV was used to perform this operation.

A listing of the FORTRAN program designed for the traditional extended Kalman filter is contained in Appendix C.

V. TESTING AND SIMULATION

Both the sequential and traditional Kalman filter routines were tested first using deterministic tracks at speeds of 5.0 to 25.0 knots and no measurement noise with a single hydrophone array. The only errors allowed in the first phase of testing were in position and velocity in the initialization of the filter.

Computer generated tracks were tested in the first series of straight running, constant depth and constant velocity torpedoes. A variety of track scenerios were used transiting through multiple quadrants including:

1. crossing north of the array
2. crossing south of the array
3. inbound to the array
4. outbound from the array
5. crossing over top of the array

All runs were made with a variety of initialization errors in position and velocity.

In the second series of tests, white, zero-mean Gaussian noise was added to corrupt the observed transit times.

The noise was added to the straight running, constant depth tracks. Before this series of tests could be conducted a gating scheme was designed to protect the filter from spurious erroneous time or positional data.

In the third series of tests, a number of torpedo maneuvers

were added to the target tracks. One-third, two-thirds and one-G turns were used. These tracks were tested with and without noise.

The torpedo velocities were increased to the 40 to 50 knot range in the fourth series of tests for straight running tracks with and without noise corruption. Maneuvers were then added to these higher velocity tracks.

In the last series of tests, the handoff routine described at the end of this section was added to the filters and tracks that traversed through the areas of multiple arrays were tested.

A. THE GATING SCHEME

The operation of the filter may be adversely affected by large measurement noise. One error of a relatively large magnitude could invalidate the filtered output for many subsequent time slots. Before random measurement noise and random excitations could be added to the observed times for testing, a form of protection was designed to guard against catastrophic failure. This protection is provided by establishing limits of acceptability for each of the measurements.

Measurement errors can occur because of many factors including an error in the transit time of the acoustic pulse primarily due to the receipt of multipath signals from previous time slots that have bounced off the surface, bottom or different density layers, or large errors in the estimates of position or velocity.

A three-sigma gate was designed using the covariance of measurement noise (R) and the covariance of estimation error ($P(K/K)$).

For each calculation of a state estimate ($\hat{X}(K/K)$), the largest positional covariance of error was used, either X, Y or Z, and converted to time in seconds using the average velocity of sound in water for Dabob bay, 4860 ft/sec. The gate then was written for each time measurement $i = 1$ to 4:

$$\text{GATE} = \sqrt{\frac{P(K/K)_{\text{largest}}}{4860.} + R_{ii}}$$

The gate expands or decreases depending on the confidence level of the transit time and position estimate. If ZDIFF which is the difference between the actual transit time received and the predicted transit time to a particular hydrophone exceeds the gate, the measurement is considered unacceptable and the filter gain is set to zero causing the filter to ignore the data and take the prediction of the states as the estimate.

$$\hat{\tilde{X}}(K/K) = \hat{\tilde{X}}(K/K-1)$$

For the sequential extended Kalman filter, because of the iterative aspect, a large erroneous time measurement zeros only the gain column for that particular hydrophone causing only that hydrophone's data to be ignored. In the traditional one-step filter, an erroneous input from any hydrophone zeros the entire gain matrix causing the filter to ignore data for the whole time slot.

B. MULTIPLE ARRAY TRACKING

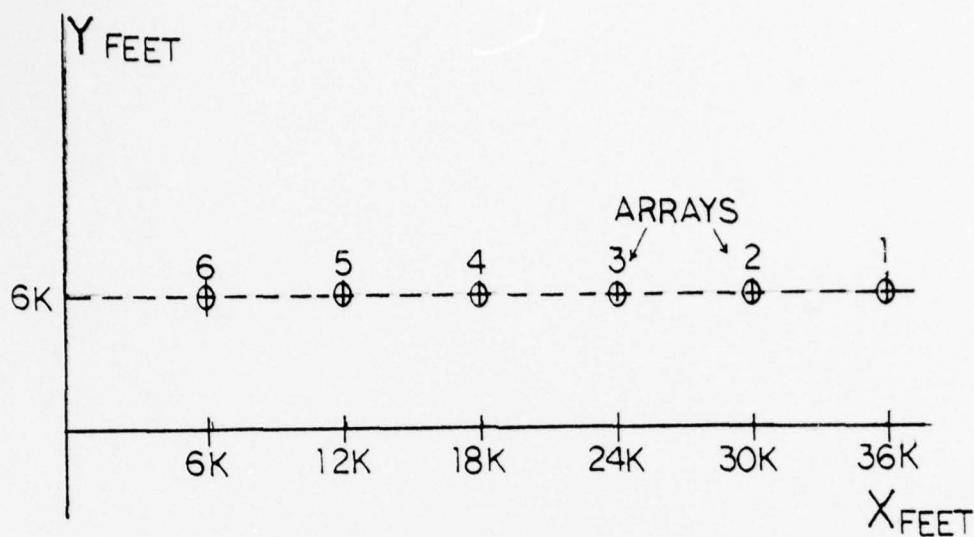
Initial tests were performed on tracks in the area of one array. In order to more closely simulate a typical run on the range, a scheme was designed to track a target through multiple arrays.

First, a coordinate system was defined as shown in Figure 2. The center of the coordinate system is geographically near the entrance to Dabob bay in the simulation. Array number 6 is the closest array to the coordinate center. Each hydrophone in a particular array has an X, Y, Z position. In the simulation array 1 was at 36,000 feet from coordinate center and array 6 was 6000 feet. The C hydrophone was assumed to be the axis location of each array. Then each X position for the X hydrophone in each array was X_C+30 , each Y position for the Y hydrophone was Y_C+30 and each Z position for the Z hydrophone was Z_C+30 . These 72 positions, an XYZ position for each of 4 hydrophones in 6 arrays, were placed into a 6x12 matrix HYDRO and referenced throughout the routine. The geometry centered on each array was taken out of the problem and the target position was based on a central reference.

The non-linear time equation became:

$$T = 1/VEL \sqrt{(X-X_0)^2 + (Y-Y_0)^2 + (Z-Z_0)^2}$$

where X_0 , Y_0 or Z_0 is the position of a particular hydrophone and array being used. In the filter routine X, Y and Z were the predicted positions $\hat{X}(K/K-1)$, and the time equation was



Coordinate System for Multiple Array Tracking

| | C Hydro | | | X Hydro | | | Y Hydro | | | Z Hydro | | |
|---|---------|------|---|---------|------|---|---------|------|---|---------|------|----|
| | X | Y | Z | X | Y | Z | X | Y | Z | X | Y | Z |
| 1 | 36000 | 6000 | 0 | 36030 | 6000 | 0 | 36000 | 6030 | 0 | 36000 | 6000 | 30 |
| 2 | 30000 | 6000 | 0 | 30030 | 6000 | 0 | 30000 | 6030 | 0 | 30000 | 6000 | 30 |
| 3 | 24000 | 6000 | 0 | 24030 | 6000 | 0 | 24000 | 6030 | 0 | 24000 | 6000 | 30 |
| 4 | 18000 | 6000 | 0 | 18030 | 6000 | 0 | 18000 | 6030 | 0 | 18000 | 6000 | 30 |
| 5 | 12000 | 6000 | 0 | 12030 | 6000 | 0 | 12000 | 6030 | 0 | 12000 | 6000 | 30 |
| 6 | 6000 | 6000 | 0 | 6030 | 6000 | 0 | 6000 | 6030 | 0 | 6000 | 6000 | 30 |

HYDRO -- Hydrophone Location Matrix

FIGURE 2

used to calculate the estimate of the measurement times $\hat{M}(\underline{Y})$. The decision parameter used to determine the switching from array to array was a straight handoff. If the predicted X position was greater than 3000 feet from the array in use, then an index (I8) was incremented and the next row of HYDRO was implemented. This placed into the routine the $X_0 Y_0 Z_0$ positions of the hydrophones in the next array. The handoff can easily be utilized in real range operations, as the transit times from adjacent arrays are present at the computer for a particular time slot.

For simulation, it was assumed that in all the arrays each axis pointed in the same direction. In range operations, the positions of the particular hydrophones referenced to the central coordinate system can be input into the matrix HYDRO to correct for OFF AXIS discrepancies.

VI. SIMULATION RESULTS

A. SERIES ONE

This series of tests included straight running, constant depth, constant velocity tracks with no noise. Various target speeds were tested ranging from 5.0 to 25.0 knots.

The only induced errors in this series of tests were in initial target position and velocity. Both the sequential and traditional filter routines effectively handled initial position errors in the X AND/OR Y direction from 0 to 25 feet and initial velocity errors from 0 to 10 ft/sec. The filter estimate was within 3 feet and 1 ft/sec in a maximum of 3 time slots. In a number of worst case tests, initial position errors of up to 50 feet and initial velocity errors of 60 ft/sec and 80 ft/sec were used. Both filter routines had the estimate on track within seven time slots.

Figure 3 is a geographical plot of a typical series one test using the sequential extended Kalman filter. The initialization of the filter was 14 feet off in X and 21 feet off in Y with no error in Z. For a 25 knot target initial velocity errors were 3 ft/sec. Figures 4 through 6 depict the deviation in feet between the estimated and true positions, $X_A(K) - \hat{X}_A(K/K)$, where $A = 1, 3$ or 5 .

There was no great difference in performance between the sequential and traditional filter routines in this series of tests.

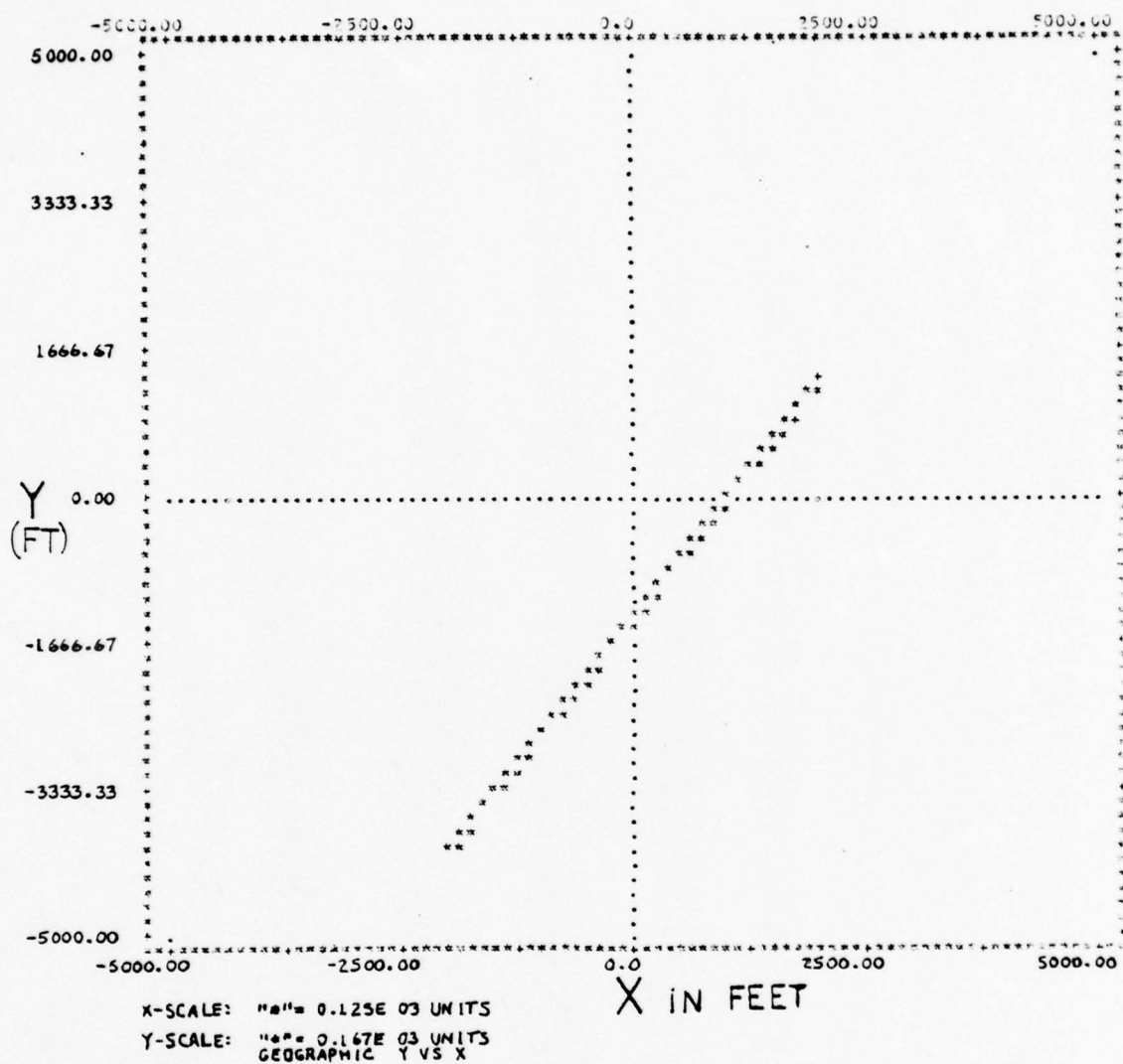


FIGURE 3: Geographic Plot Straight Running Track
in the area of a single Array

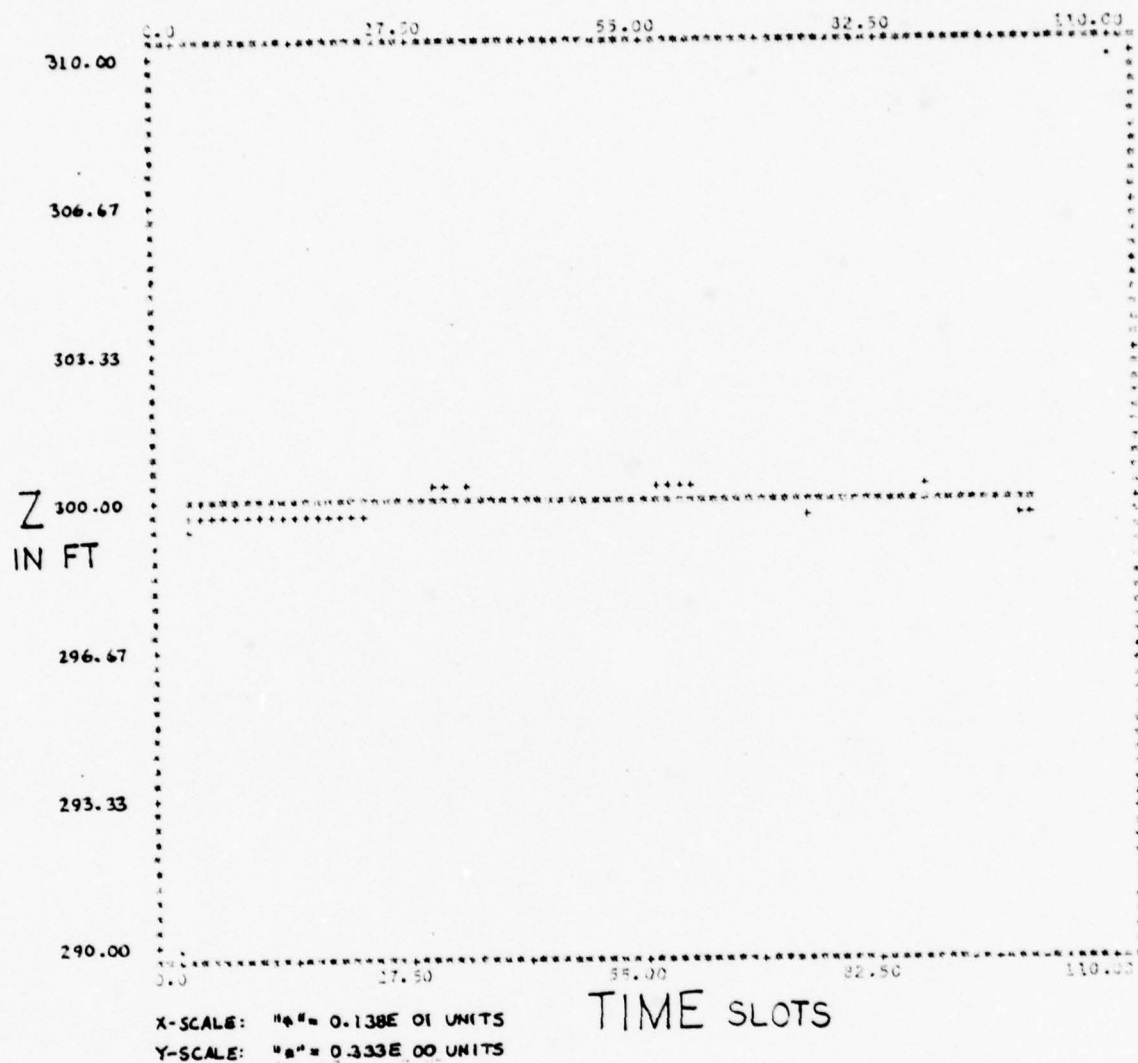


FIGURE 4: Z Position versus Time for a Straight Running
Track with No Noise in the Area of a Single Array

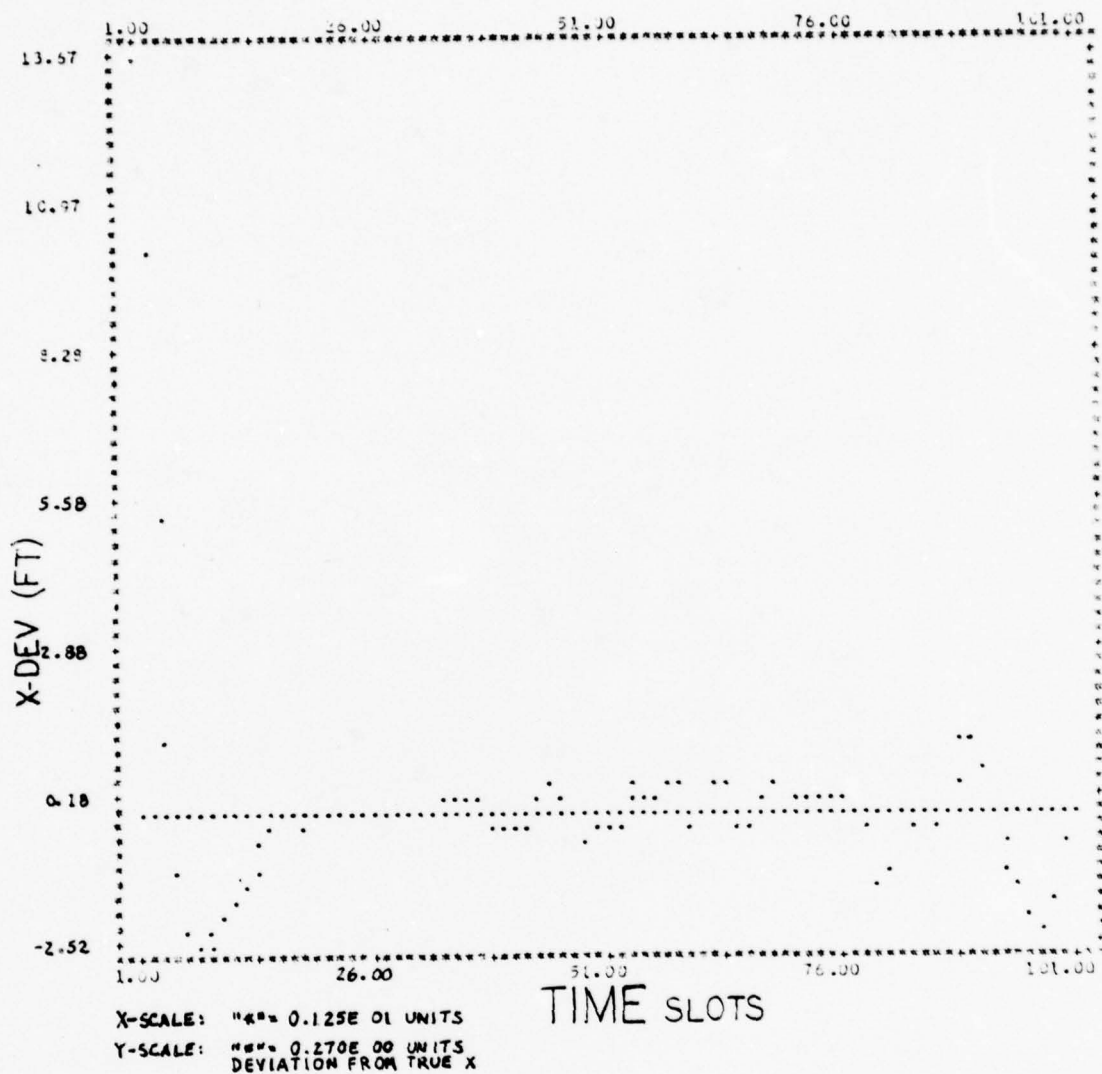


FIGURE 5: X Deviation from True versus Time for a Straight Running Track without Noise in the Area of a Single Array

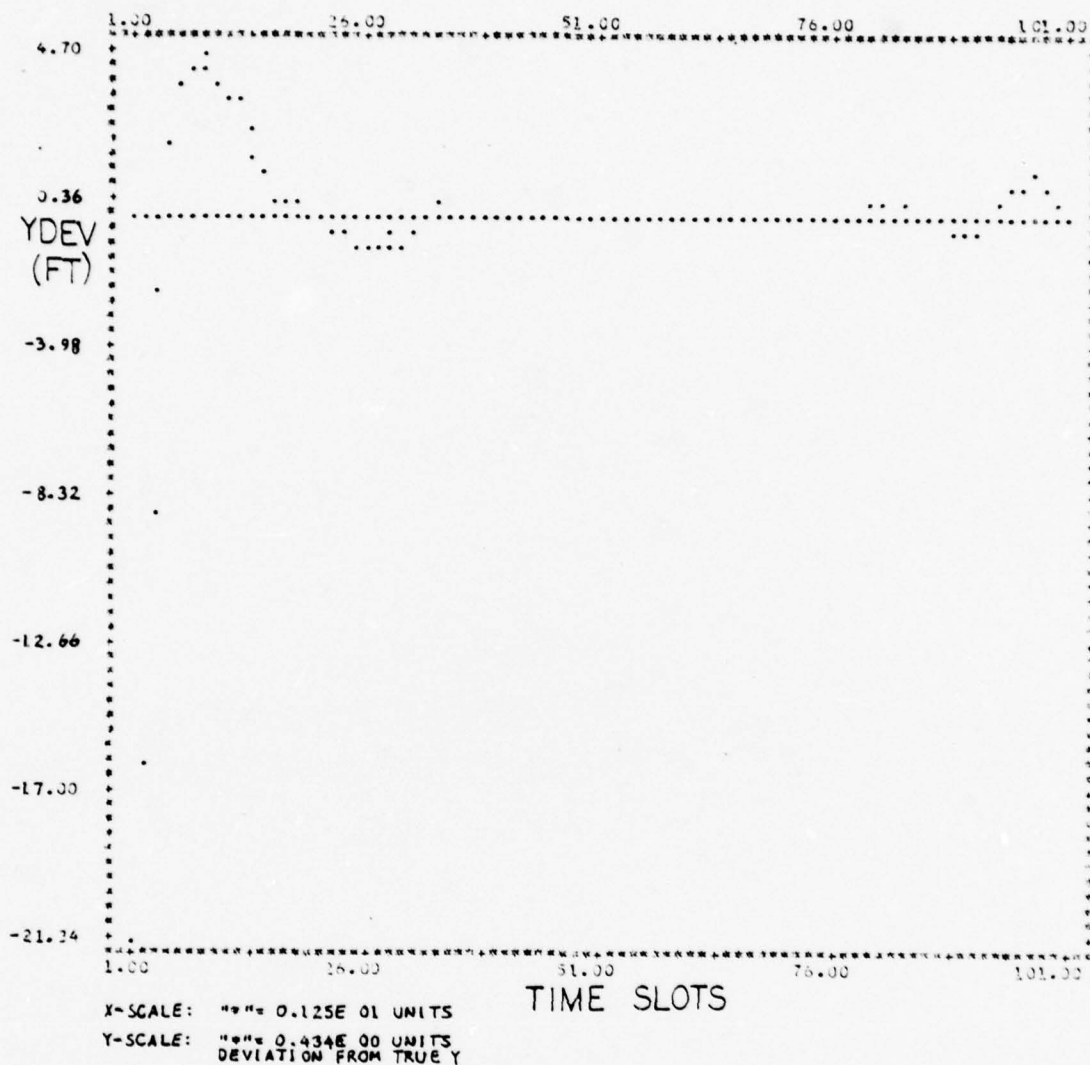


FIGURE 6: Y Deviation from True versus Time for a straight Running Track without Noise in the Area of a Single Array

B. SERIES TWO

In the second series of tests random white Gaussian noise was added to the transit time measurements for straight running tracks. Target velocities varied from 5.0 to 25.0 knots. Initial position errors ranged from 0 to 25 feet and initial velocity errors from 0 to 10 ft/sec.

Both the sequential and traditional routines performed well against the random noise. Estimate deviations from true track in X and Y positions were not observed greater than 3 feet after lock-on was acquired. Figures 7 and 8 depict the estimate deviation from true track for the sequential routine. Initial positions were 9 feet off in X, 13 feet in Y and 2 feet in Z.

C. SERIES THREE

In this series of tests maneuvers were added to the torpedo tracks and the filter was tested with and without noise.

Figure 9 is a geographical plot of a 25 knot target in a noiseless environment with 1/3-G turns at time slots 25 and 65. Figure 10 depicts the Z position versus the time of run. In the initialization of the filter X was off 13 feet and Y was off 21 feet. Figure 11 depicts the deviation by the estimate from the true track in the X direction, for the traditional inversion approach. Figure 12 shows the deviation for the same track using the sequential extended Kalman filter. When the target initiated the turn at time slots 25 and 65, the

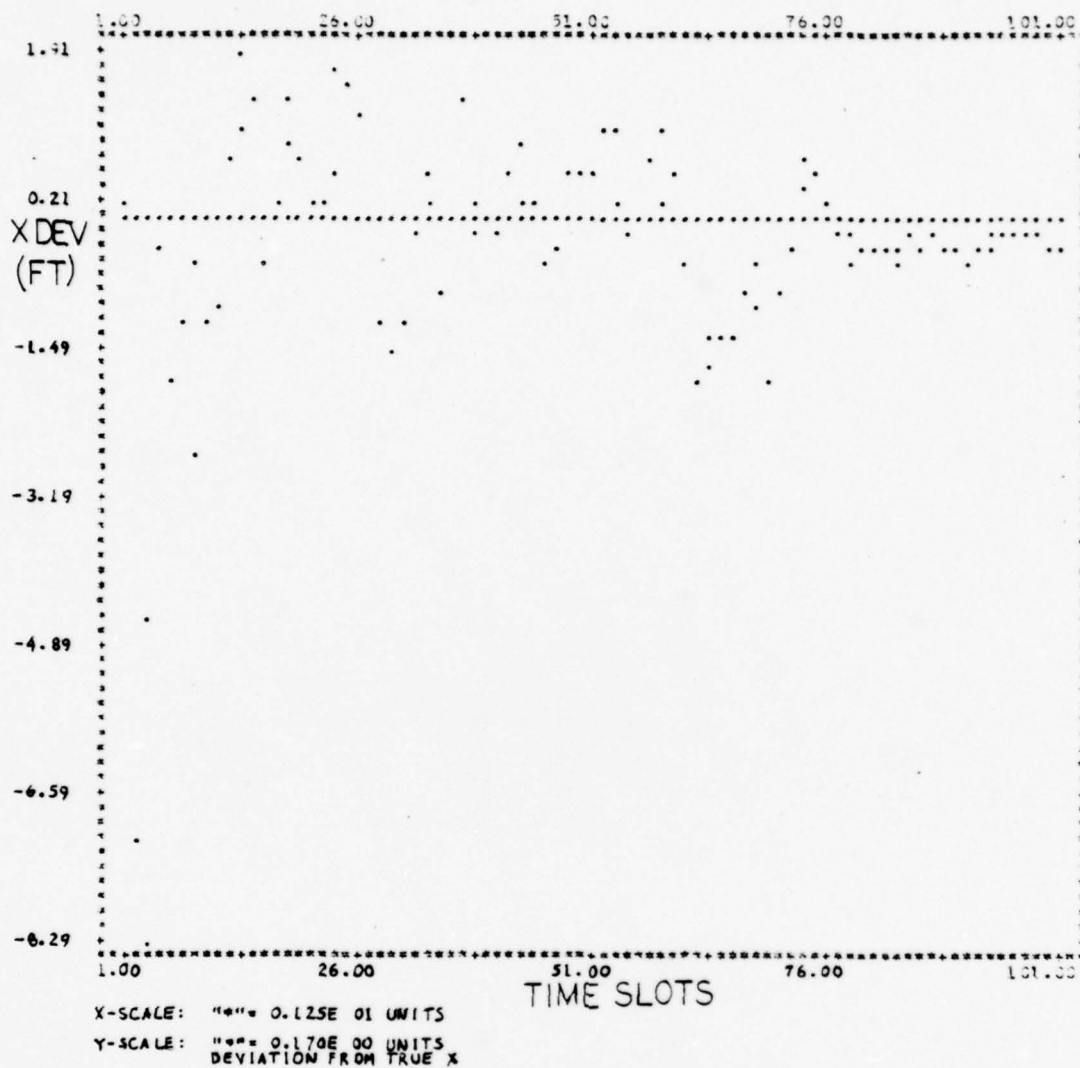


FIGURE 7: X Deviation from True versus Time for a straight Running Track with Noise in the Area of a Single Array -- Sequential Routine

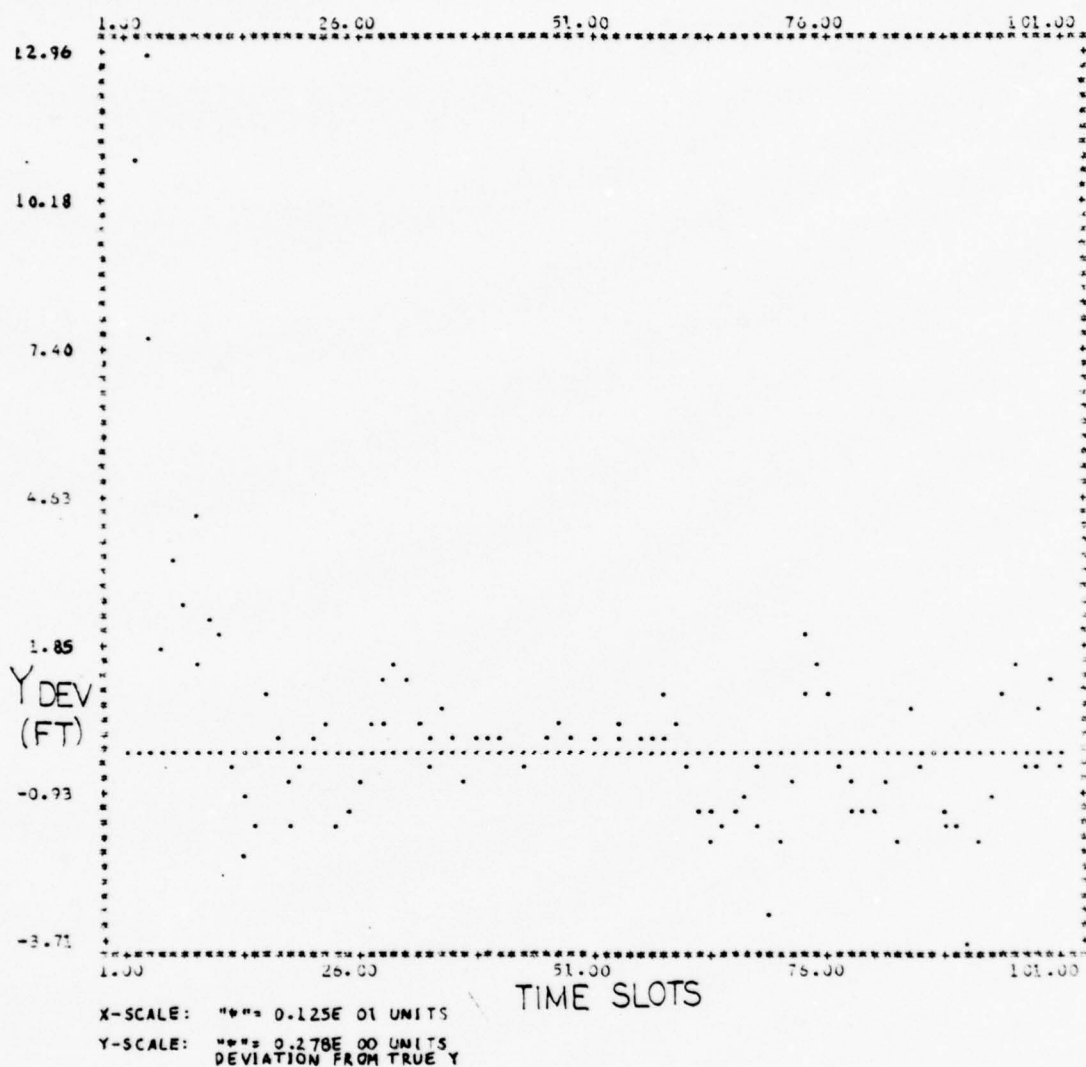


FIGURE 8: Y Deviation from True versus Time for a straight Running Track with Noise in the Area of a single Array -- Sequential Routine

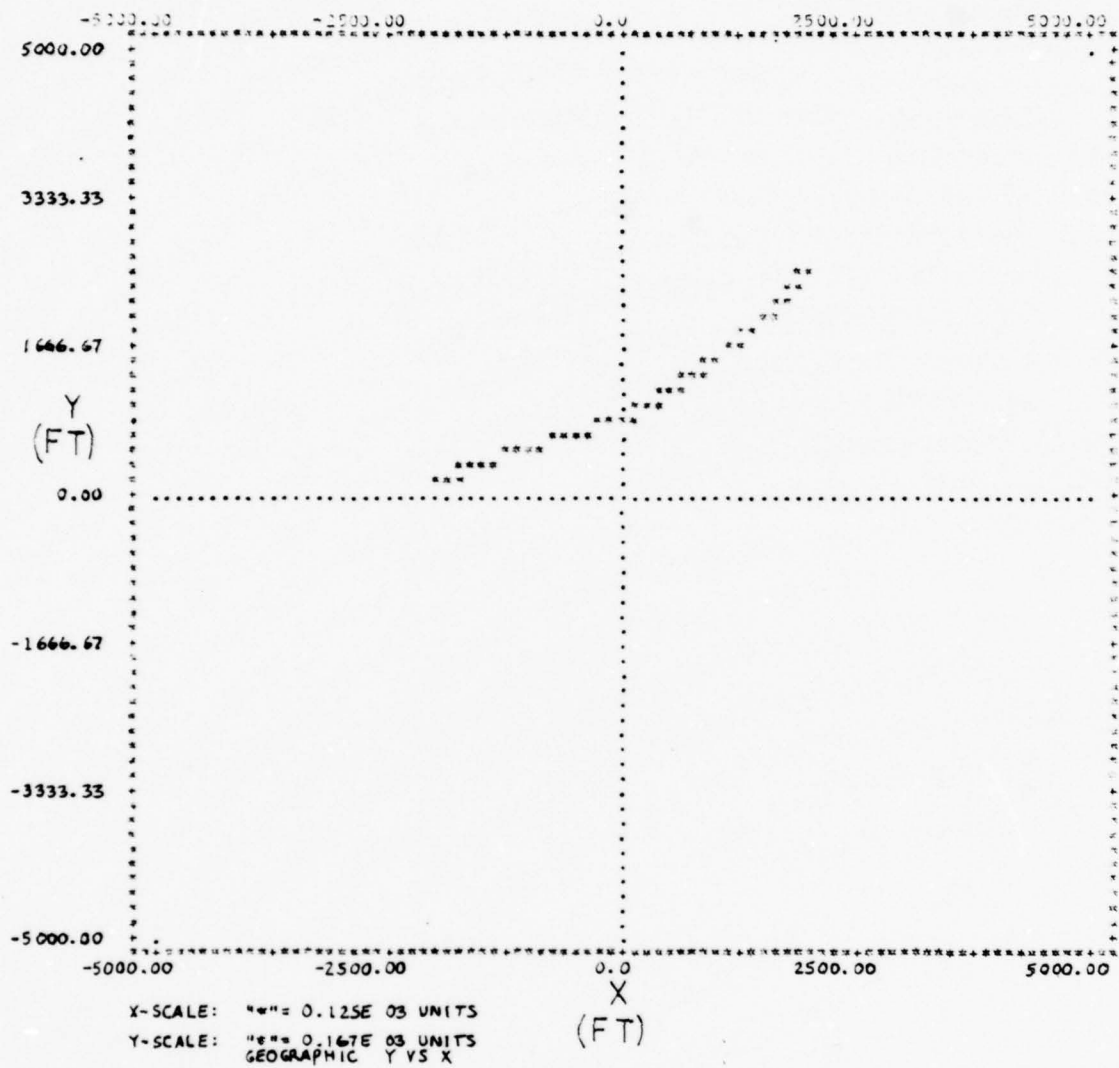


FIGURE 9: Geographic Plot of a Track with 1/3-G
Turns in the Area of a Single Array

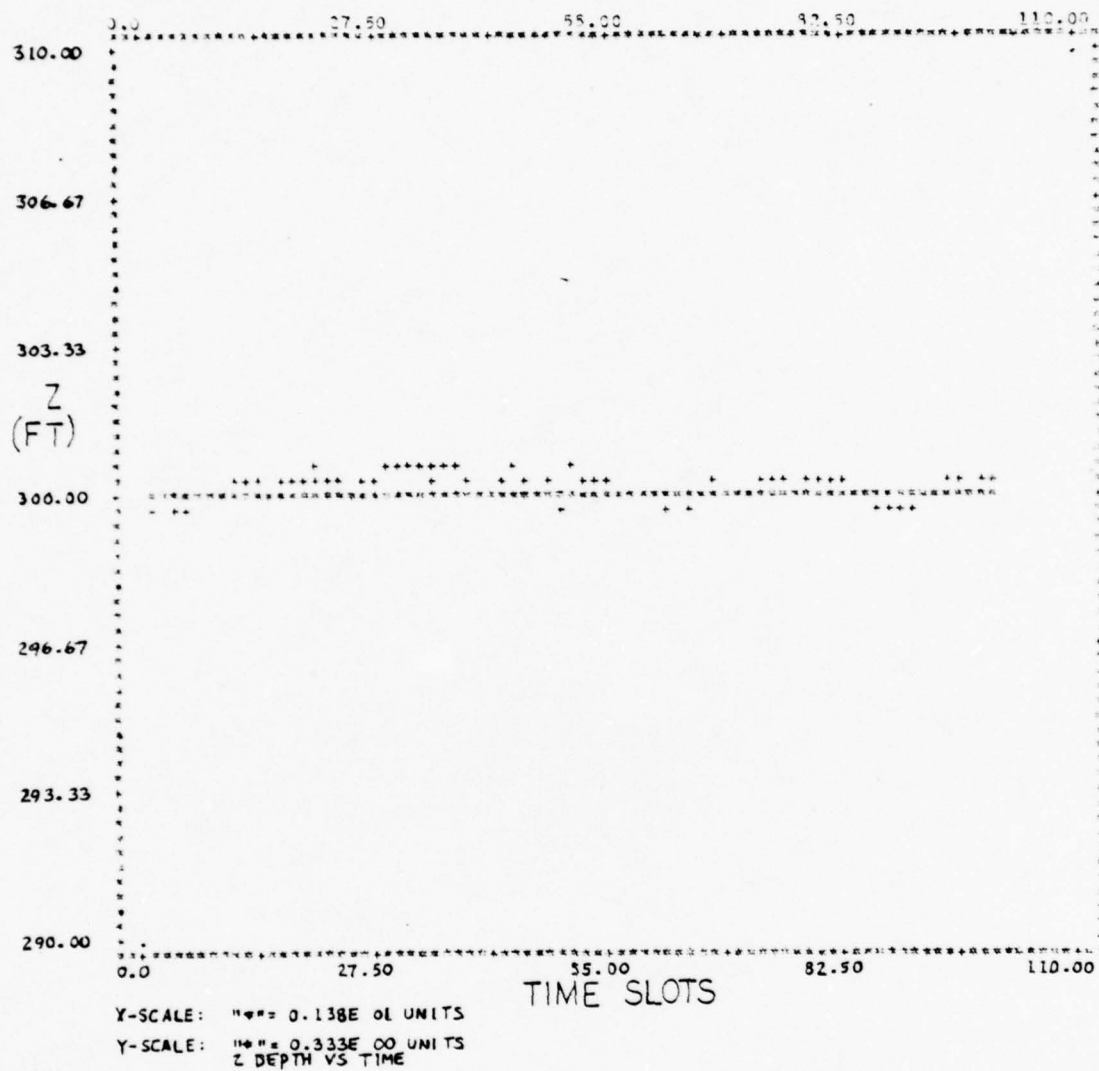


FIGURE 10: Z Position versus Time for a Track with 1/3-G Turns without Noise in the Area of a single Array

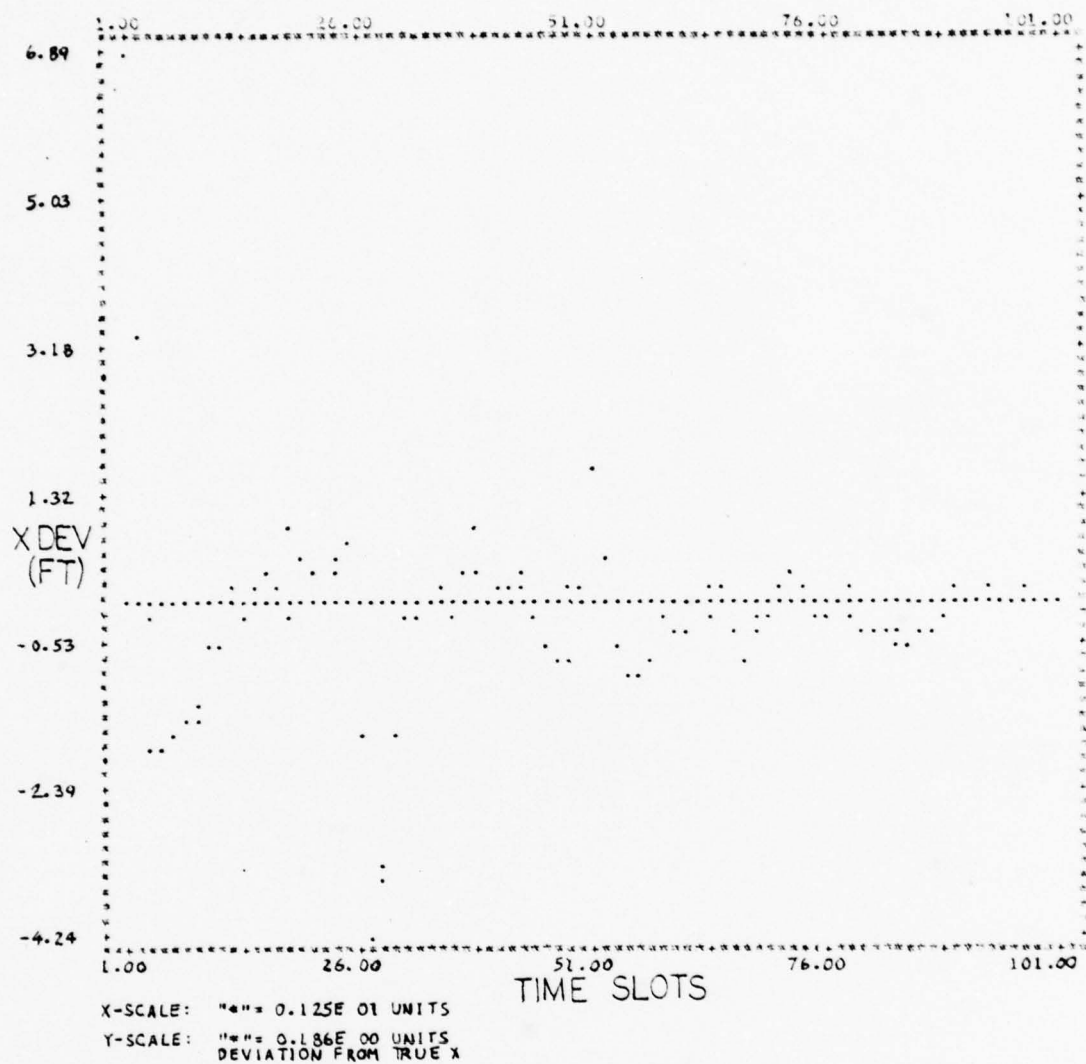


FIGURE 11: X Deviation from True for a Track with 1/3-G
 Turns without Noise in the Area of a Single
 Array -- Traditional Routine

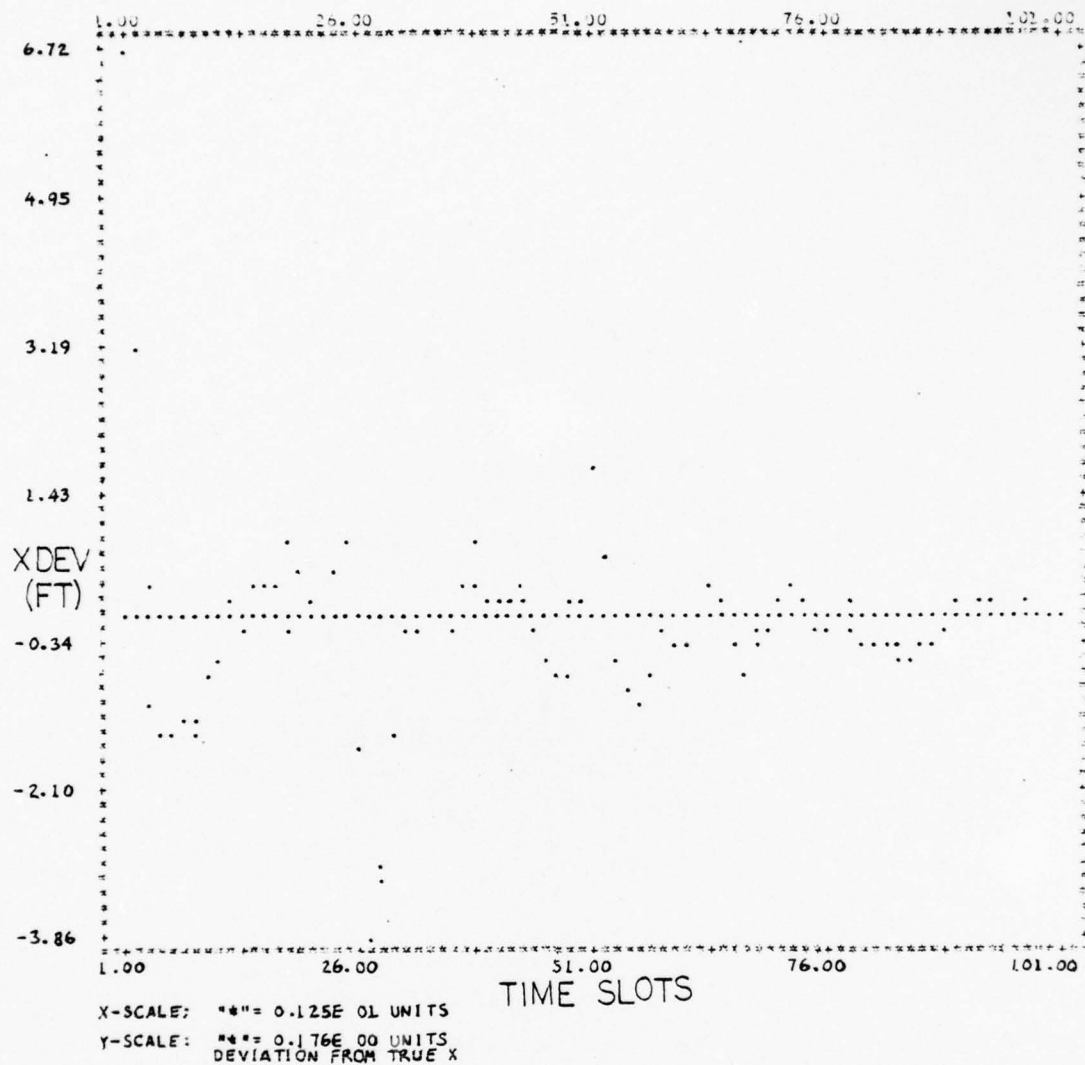


FIGURE 12: X Deviation from True for a Track with 1/3-G
Turns without Noise in the Area of a Single
Array -- Sequential Routine

covariance of estimation error and thus the filter gains increased allowing more weight to be placed on the incoming data (transit times). In this manner the filter was able to adjust the velocities in the X and Y directions and force the estimate back on track. Results from both routines were comparable as is shown in Figures 11 and 12. The routine using the sequential extended Kalman filter deviated a maximum of four feet at the turn and was back on track within 3 time slots, while the traditional filter routine had a slightly greater deviation and required 2 more time slots to acquire lock again. Figure 13 shows the deviation of the estimate from true track in the Y direction for this same run for the sequential filter and Figure 14 for the traditional filter. Both estimates deviated a maximum of 3 feet at the turn and re-acquired lock-on in 4 time slots with the sequential routine reacting slightly better to the turns.

Next, target tracks with $2/3$ and 1-G turns were tested. Figure 15 is a geographic plot of this track with a 1-G turn at time slot 25 and a $2/3$ -G turn at time slot 65. Figure 16 depicts the Z position vs time for this track. With an initialization error of 13 feet in X and 21 feet in Y, the maximum deviation after lock-on was again at the turn points. Figure 17 depicts this deviation from true track in the X direction, for the sequential routine with a maximum deviation of approximately 5.5 feet for the 1-G turn. Lock-on was re-acquired in a maximum of 4 time slots. Figure 18 shows the

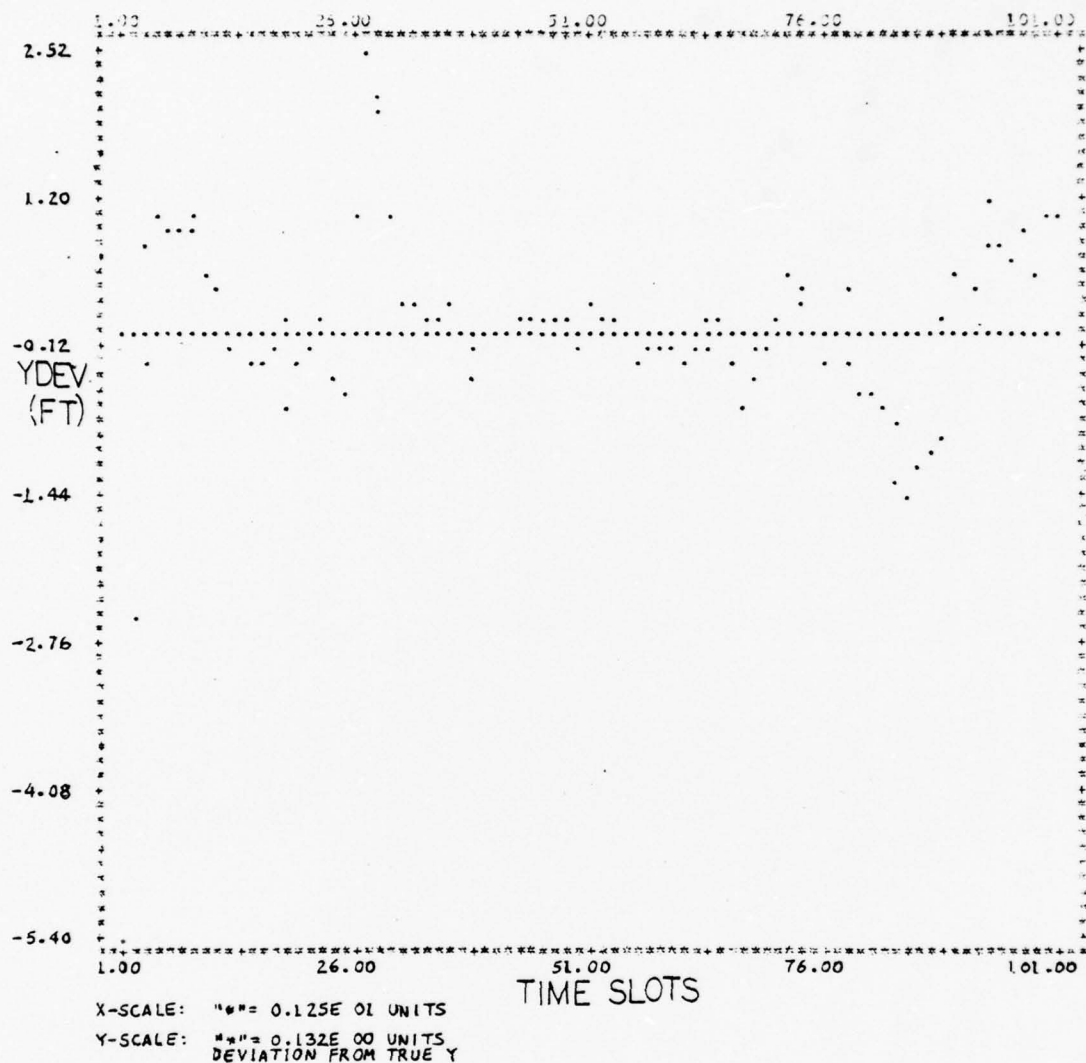


FIGURE 13: Y Deviation from True for a Track with 1/3-G
Turns without Noise in the Area of a Single
Array -- Sequential Routine

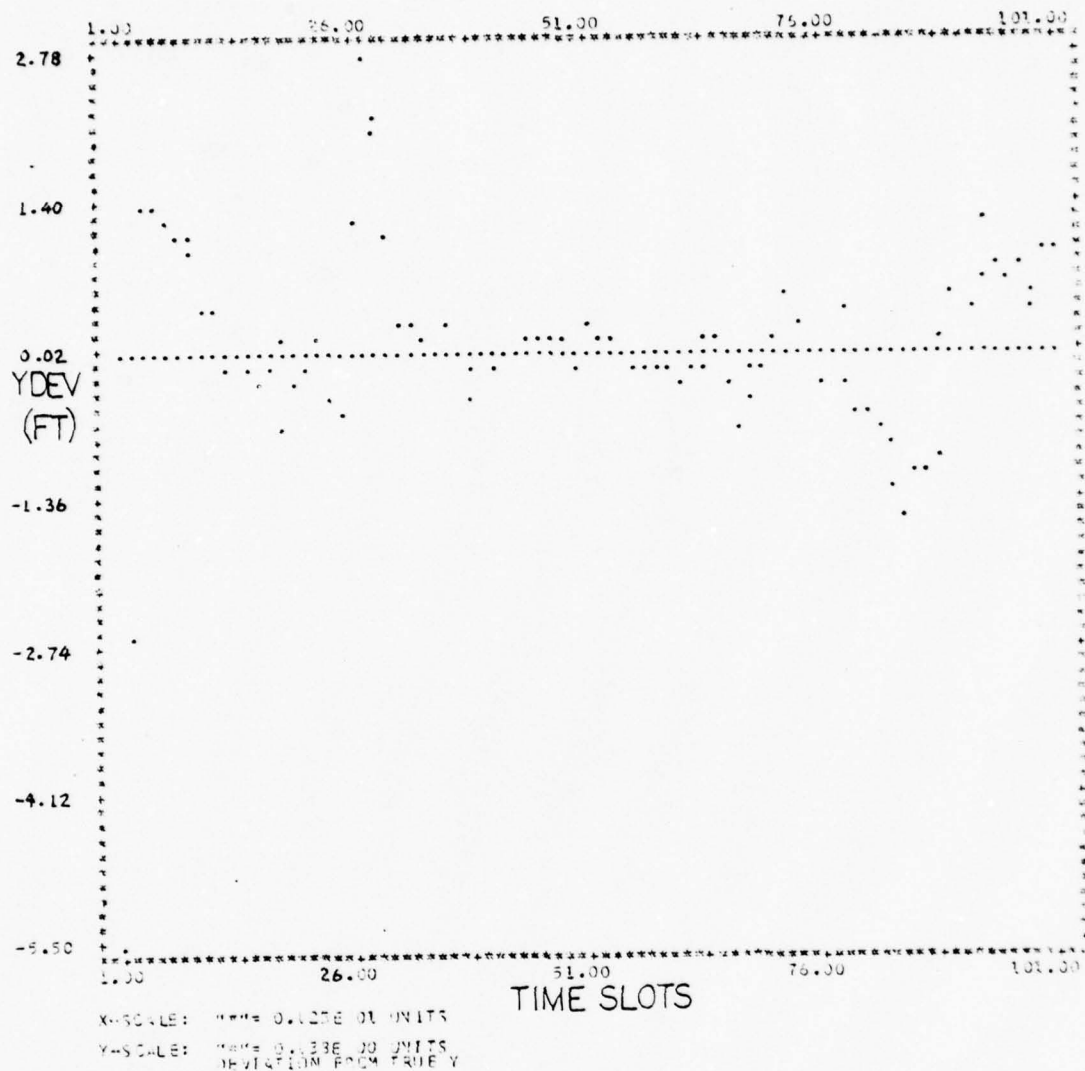


FIGURE 14: Y Deviation from True for a Track with 1/3-G Turns without Noise in the Area of a Single Array -- Traditional Routine

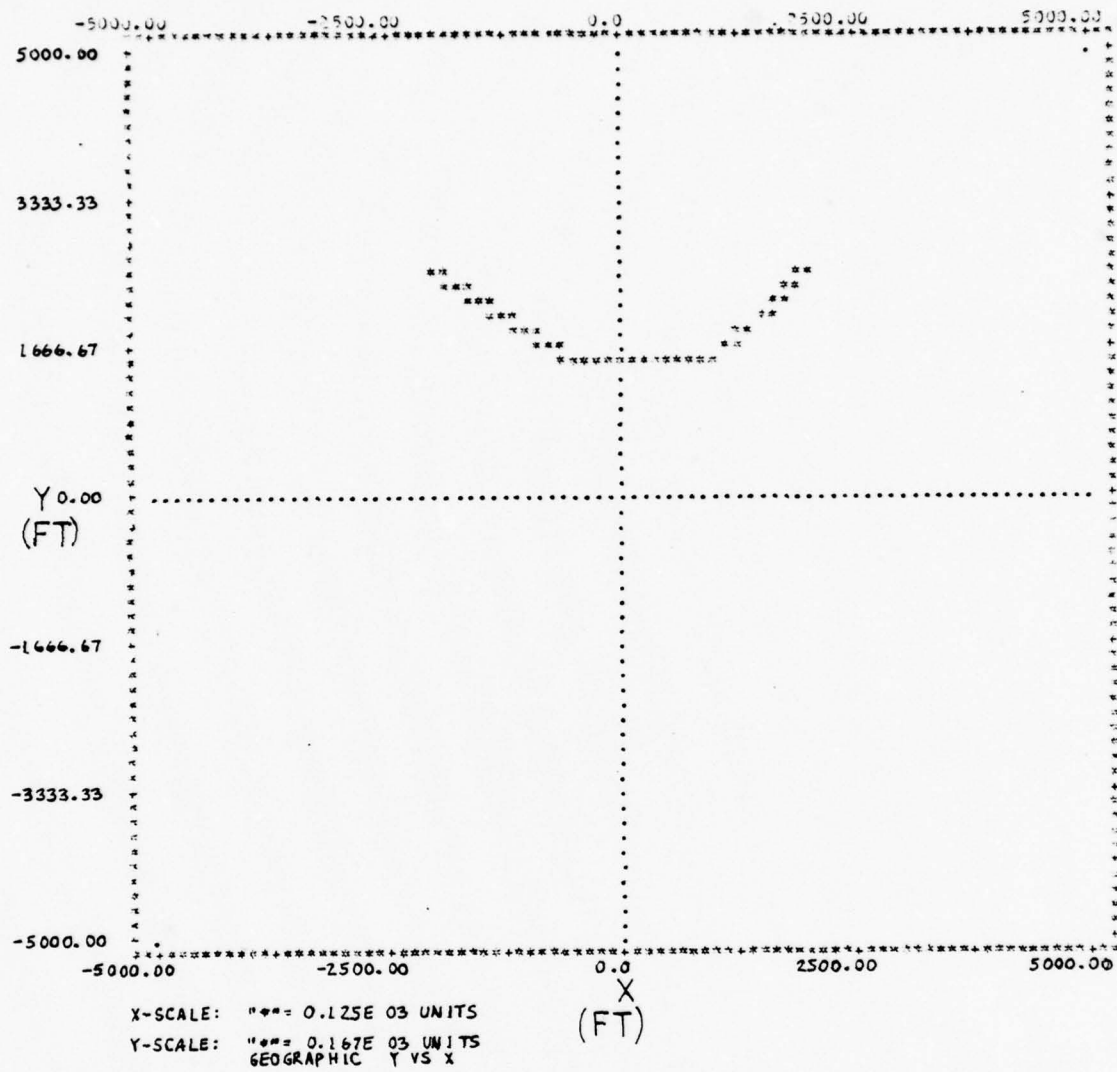


FIGURE 15: Geographic Plot for a Track with 2/3-G and 1-G Turns in the Area of a Single Array

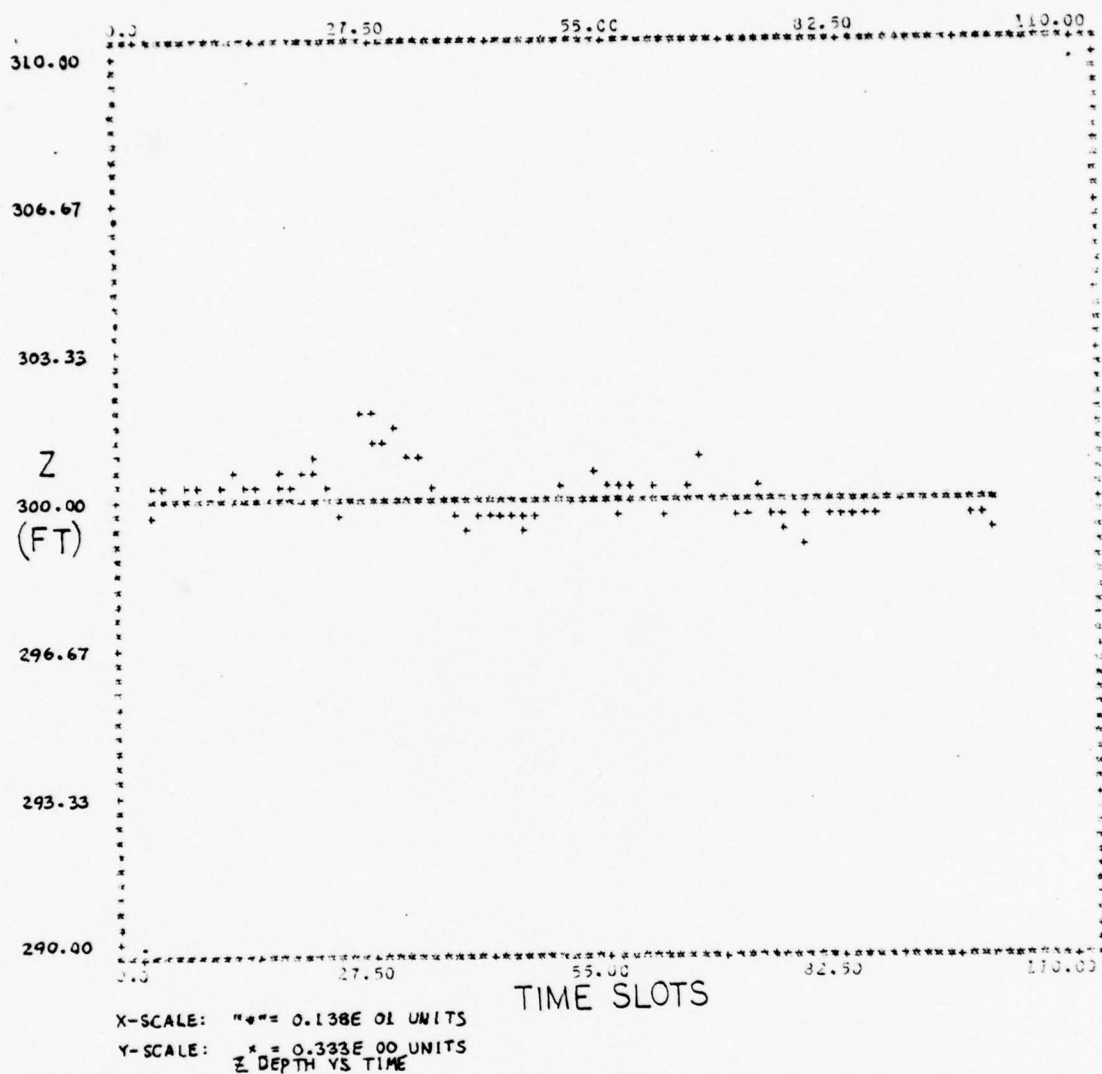


FIGURE 16: Z Position versus Time for a Track with 2/3-G and 1-G Turns without Noise in the Area of a Single Array

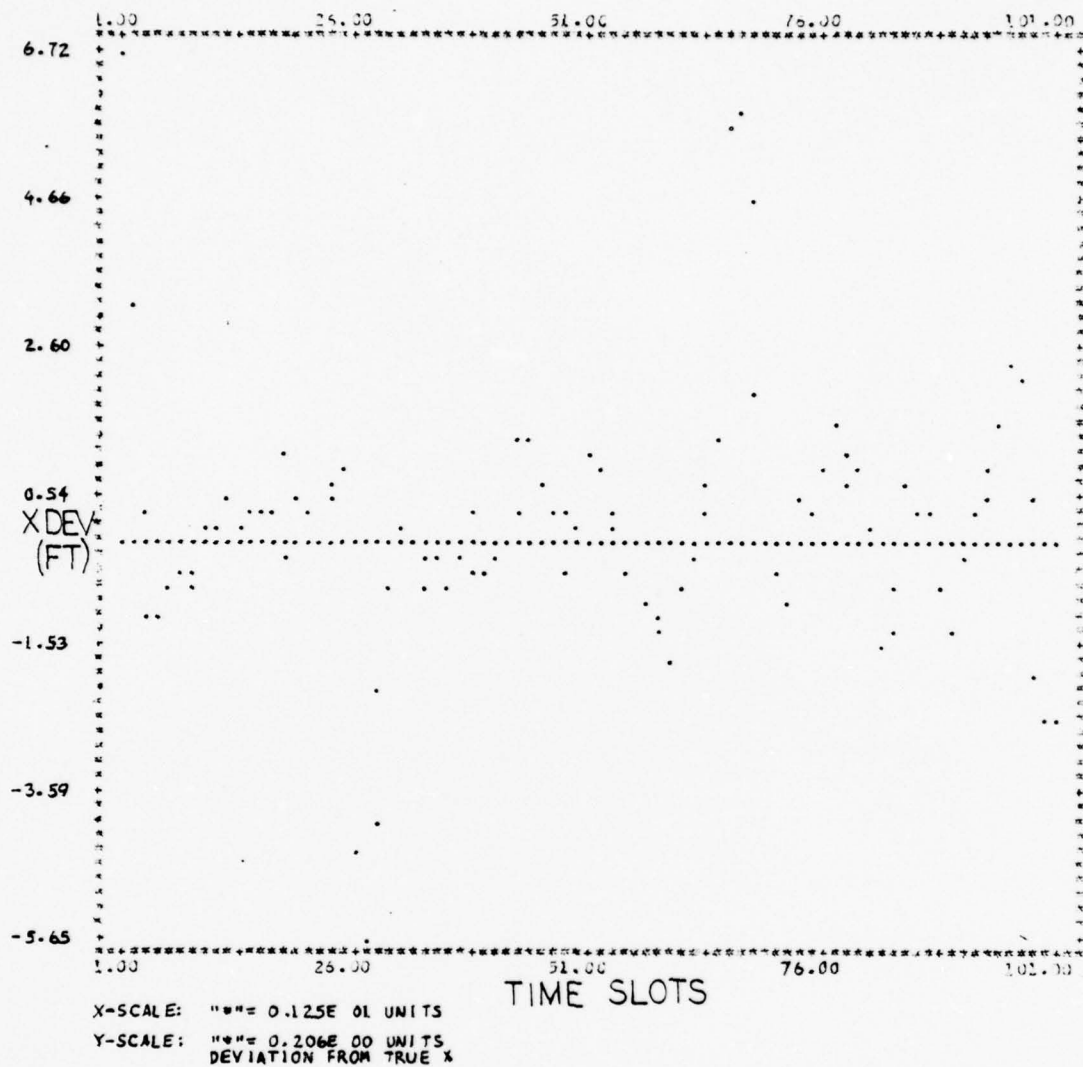


FIGURE 17: X Deviation from True for a Track with 2/3-G and 1-G Turns without Noise in the Area of a Single Array -- Sequential Routine

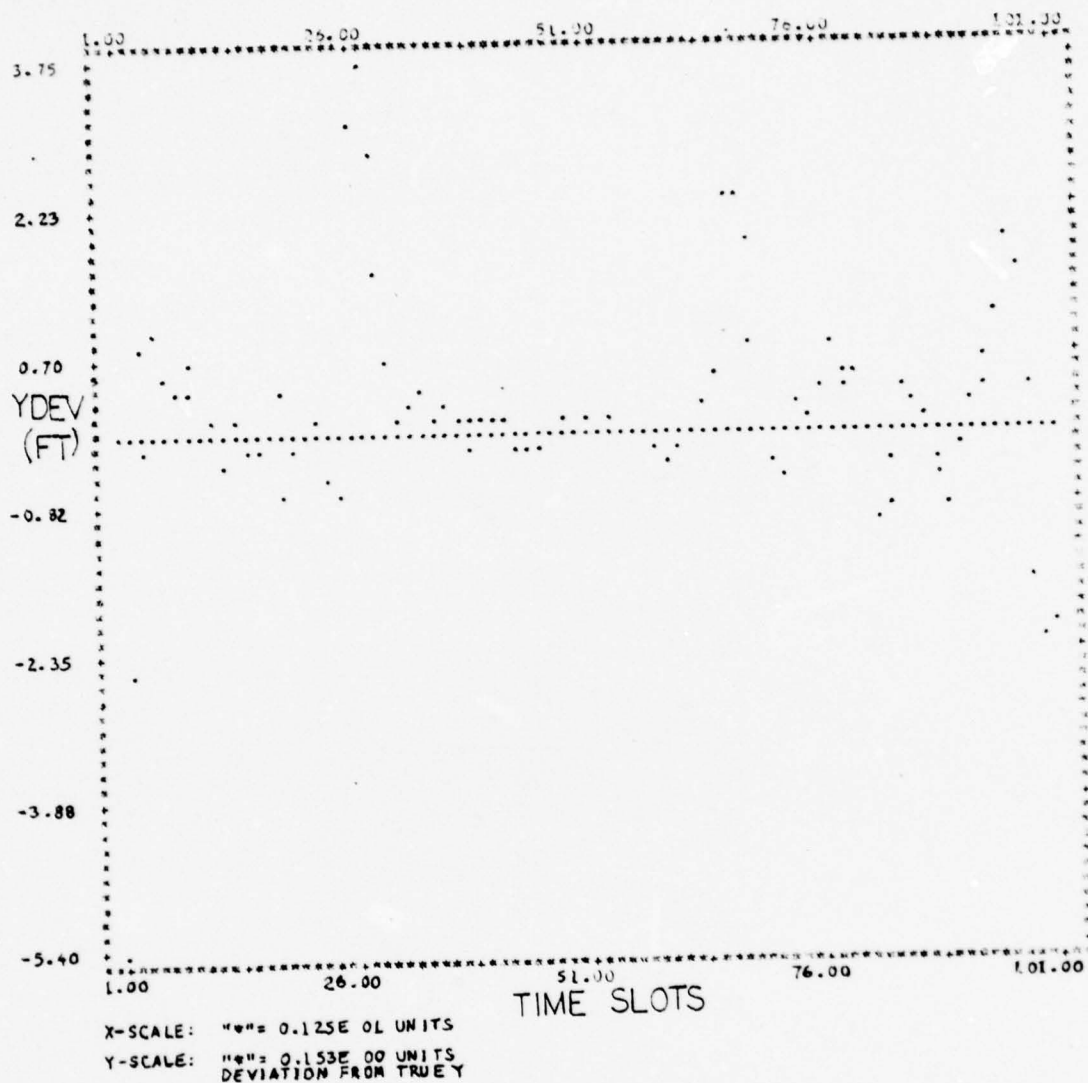


FIGURE 18: Y Deviation from True for a Track with 2/3-G and 1-G Turns without Noise in the Area of a Single Array -- Sequential Routine

deviation from true in the Y direction with a maximum deviation after lock-on for the 1-G turn of under 4 feet. Runs with the traditional extended Kalman filter routine showed slightly larger deviations with more reaction time required to re-acquire lock-on.

In the final tests for this series, zero mean, white Gaussian noise was added to corrupt the observed transit times. Figure 19 depicts the estimate's deviation from true track in the X direction using the traditional filter routine. The target experienced $1/3$ -G turns at time slots 25 and 65. Figure 20 shows the same track in noise with the same filter parameters when run with the sequential extended Kalman filter routine. Comparison shows that the latter had approximately 2 feet less maximum deviation and that while the traditional routine had trouble re-acquiring lock-on after the turn, the sequential routine regained track within 4 time slots. Figures 21 and 22 depict the same type of behavior for deviation in the Y direction. Maximum deviation again was slightly less and lock-on more efficiently re-acquired using the sequential routine.

In the cases involving more radical maneuvers, the sequential routine continued to perform better against the modeled noise. Figure 23 shows the estimate's deviation from true track in the X direction for a run with a 1-G maneuver at time slot 25 and a $2/3$ -G maneuver at time 65 in noise. This graph for the traditional routine shows maximum deviation at

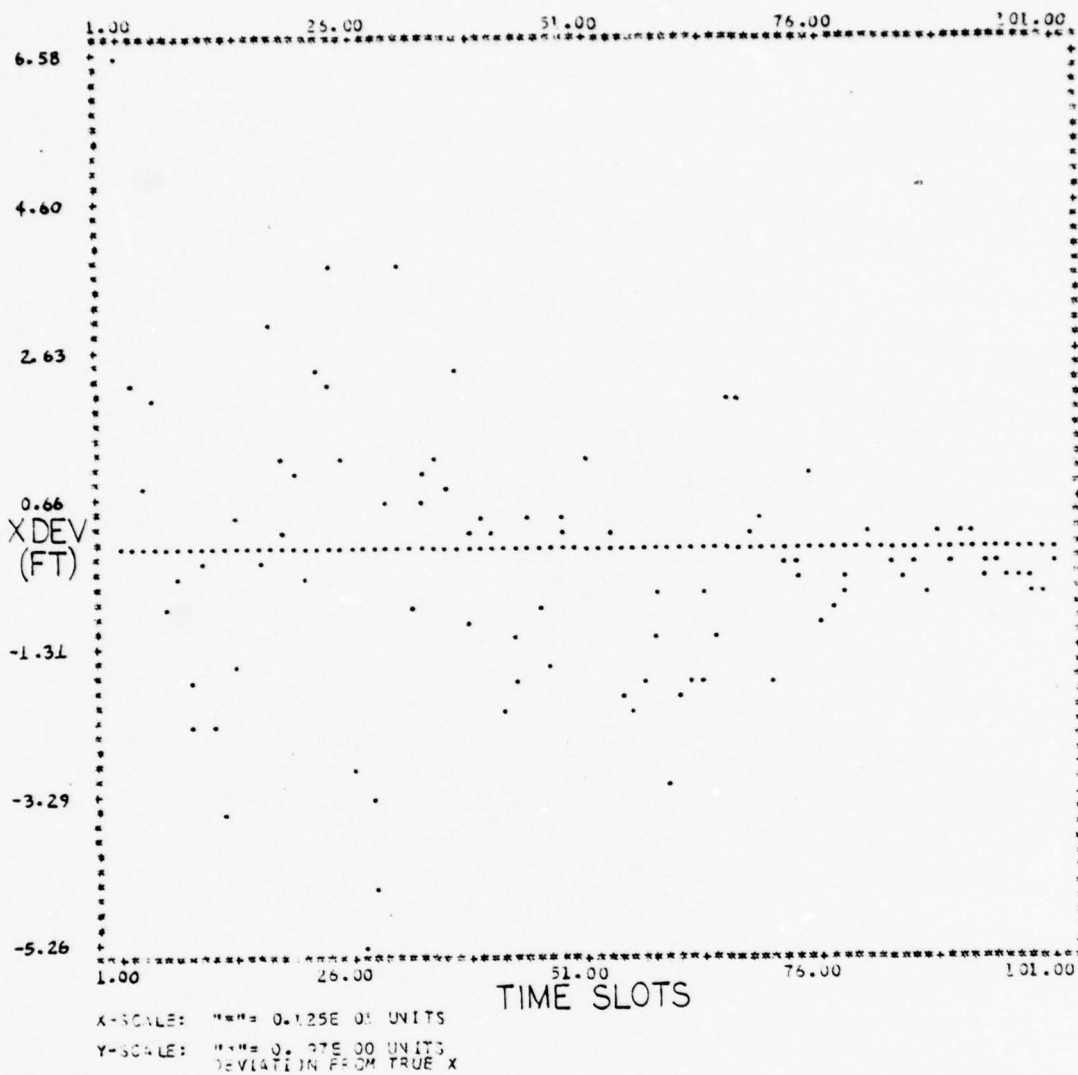


FIGURE 19: X Deviation from True for a Track with 1/3-G
Turns with Noise in the Area of a Single
Array -- Traditional Routine

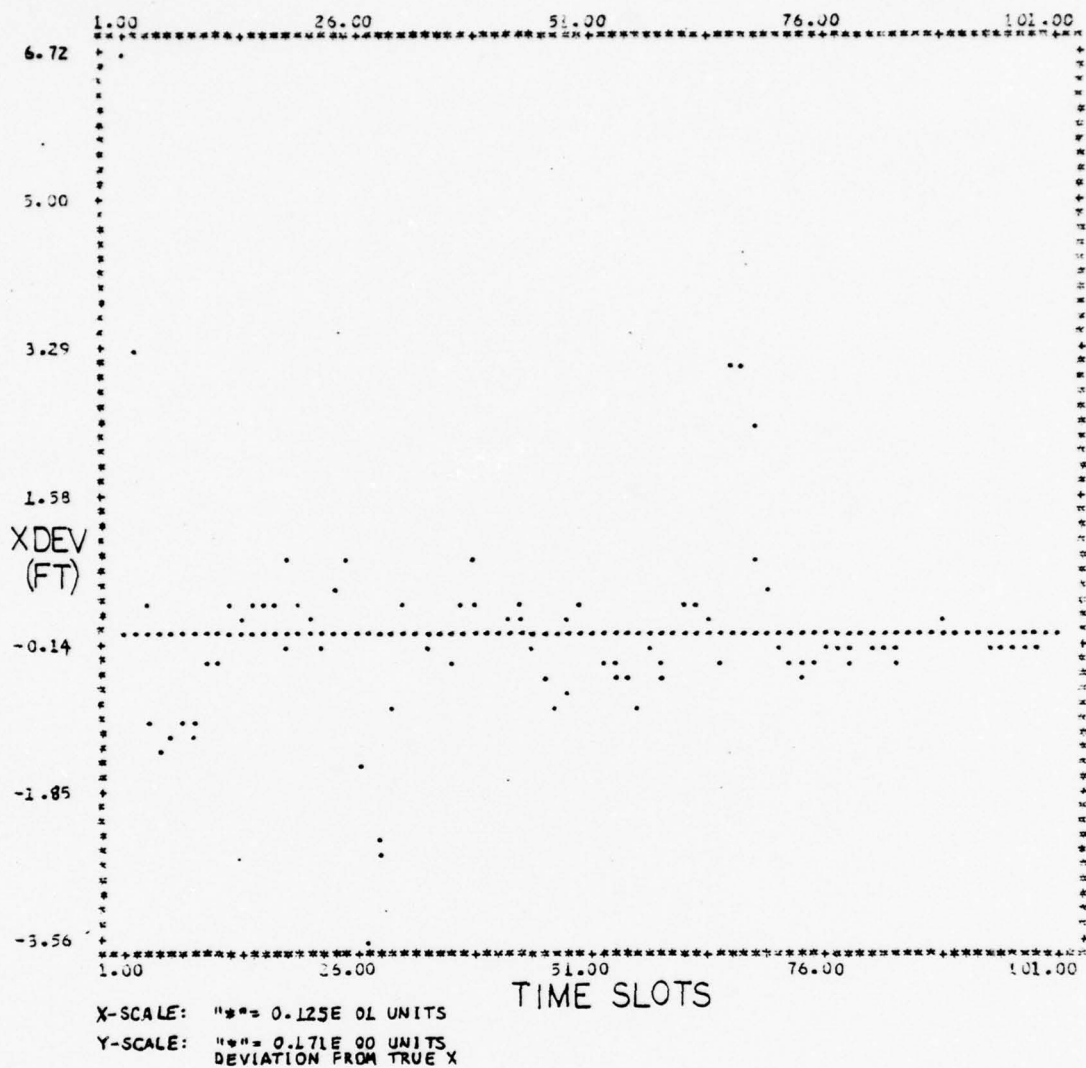


FIGURE 20: X Deviation from True for a Track with 1/3-G
Turns with Noise in the Area of a Single
Array -- Sequential Routine

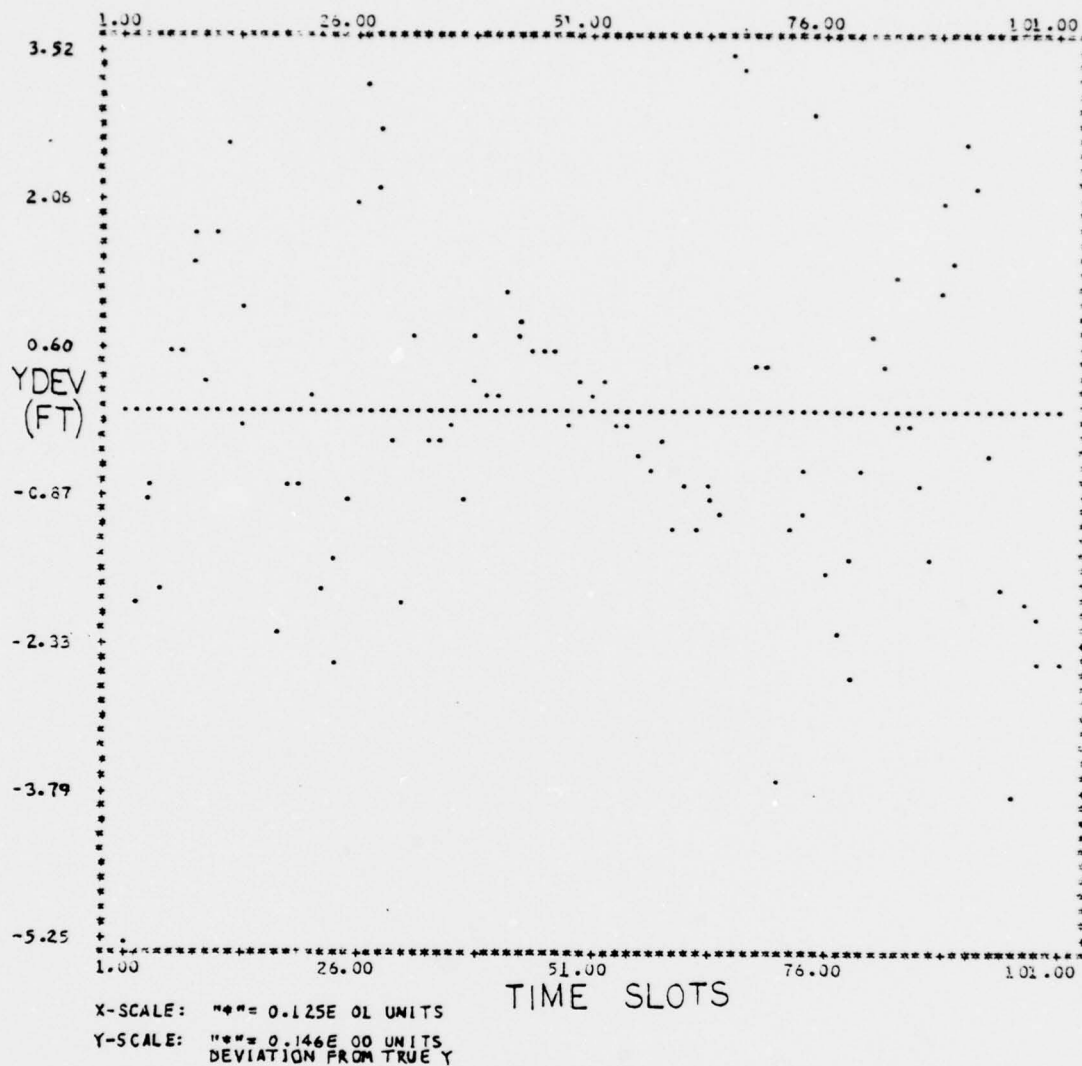


FIGURE 21: Y Deviation from True for a Track with 1/3-G
Turns with Noise in the Area of a Single
Array -- Traditional Routine

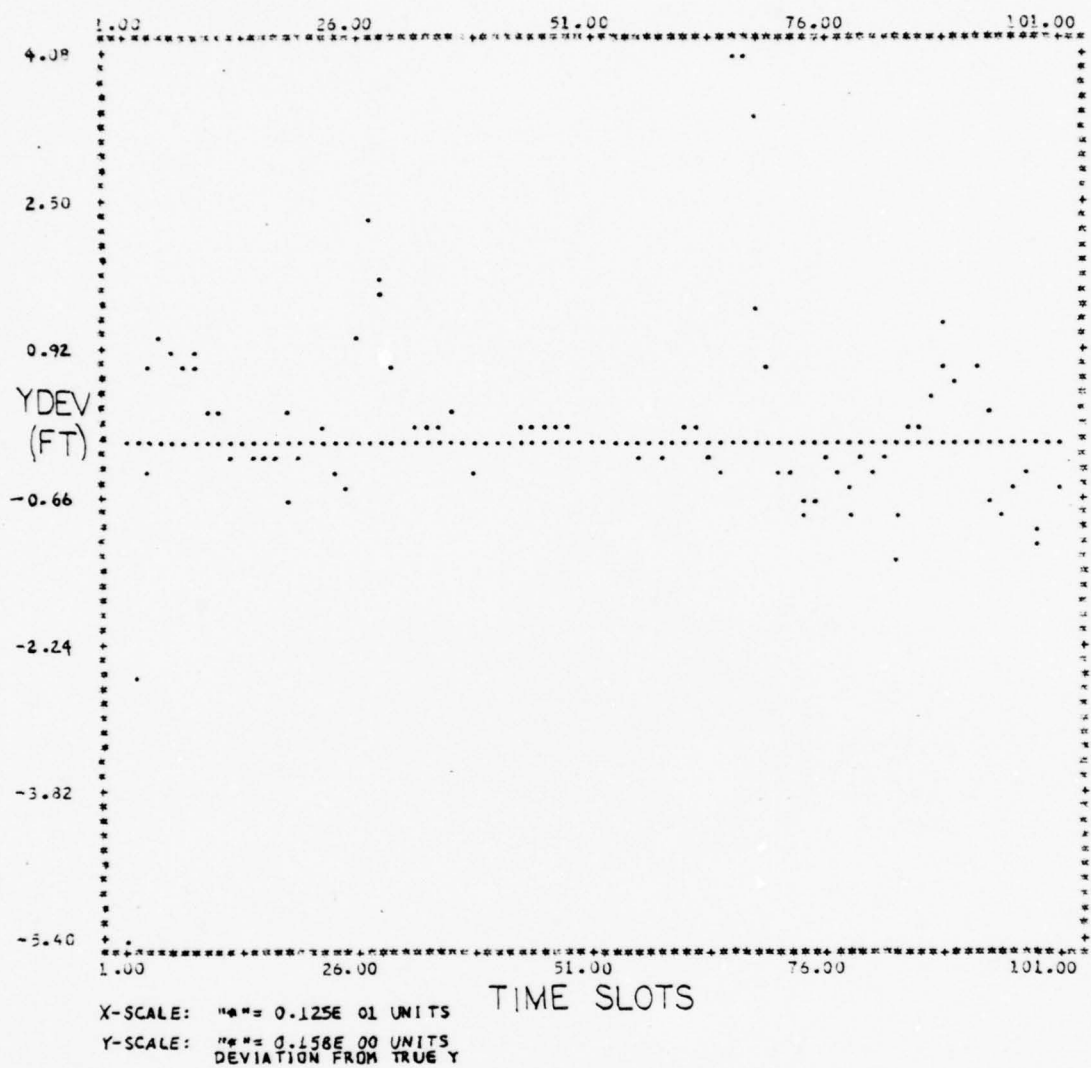


FIGURE 22: Y Deviation from True for a Track with 1/3-G
Turns with Noise in the Area of a Single
Array -- Sequential Routine

the 1-G turn of approximately 9 feet and the difficulty the filter had in re-acquiring lock-on. Figure 24, which is the same track and filter parameters run with the sequential routine, shows a marked improvement in both maximum deviation and efficiency in re-acquiring the track after a turn in noise.

Figures 25 and 26 demonstrate again the better performance of the sequential extended filter routine regarding deviation in the Y direction for the more radically maneuvering track.

D. SERIES FOUR

In the fourth series of tests, the target speeds were increased to the 40 to 50 knot range in order to bring the simulation in line with speeds actually experienced on the range. In a noiseless environment, both routines showed similar performance with a maximum deviation from true track during the run under 2 feet. With random noise added this deviation increased and Figure 27 is a plot of the deviation in the X direction using the traditional routine. When compared to Figure 28 which is the same track using the sequential routine with the same parameters, a marked improvement in maximum deviation and lock-on efficiency is noticeable. Better performance by the sequential routine was also present regarding deviation in the Y direction.

When $1/3$ -G turns were introduced, the routines performed similarly in a noiseless environment. With random noise and maneuvers the routines were again comparable with the traditional routine performing slightly better as is shown in

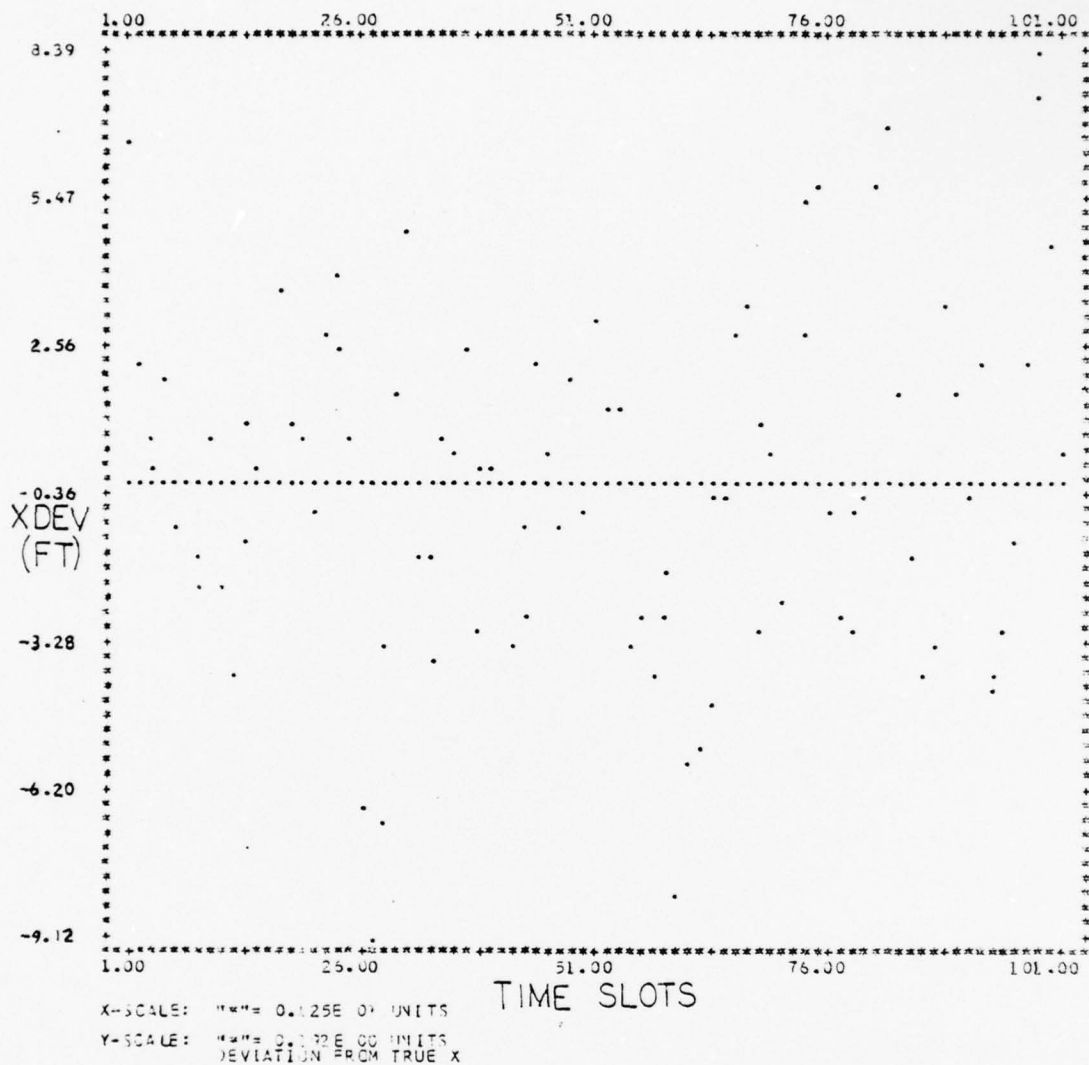


FIGURE 23: X Deviation from True for a Track with 2/3-G and 1-G Turns with Noise in the Area of a Single Array -- Traditional Routine

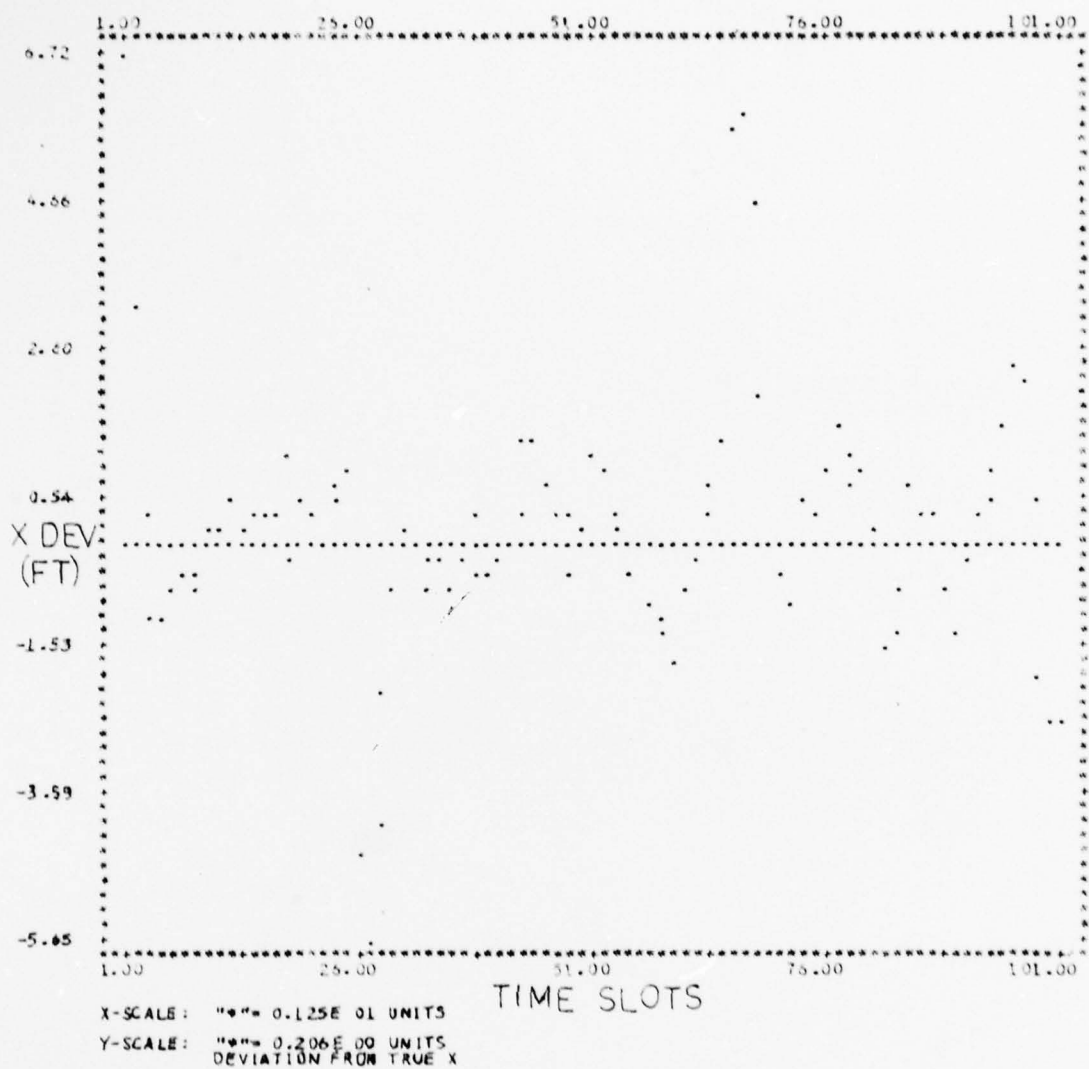


FIGURE 24: X Deviation from True for a Track with 2/3-G and 1-G Turns with Noise in the Area of a Single Array -- Sequential Routine

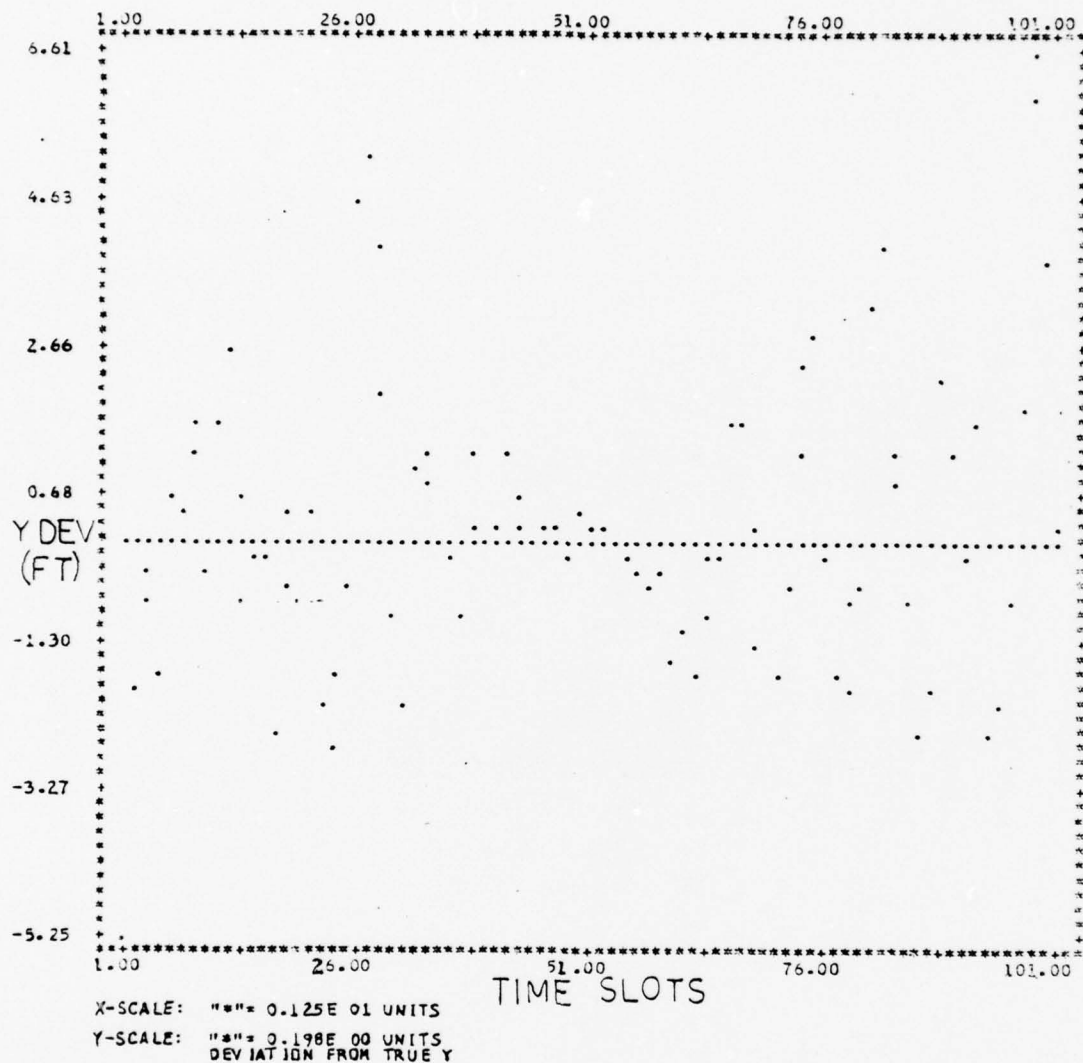


FIGURE 25: Y Deviation from True for a Track with 2/3-G and 1-G Turns in the Area of a Single Array with Noise --Traditional Routine

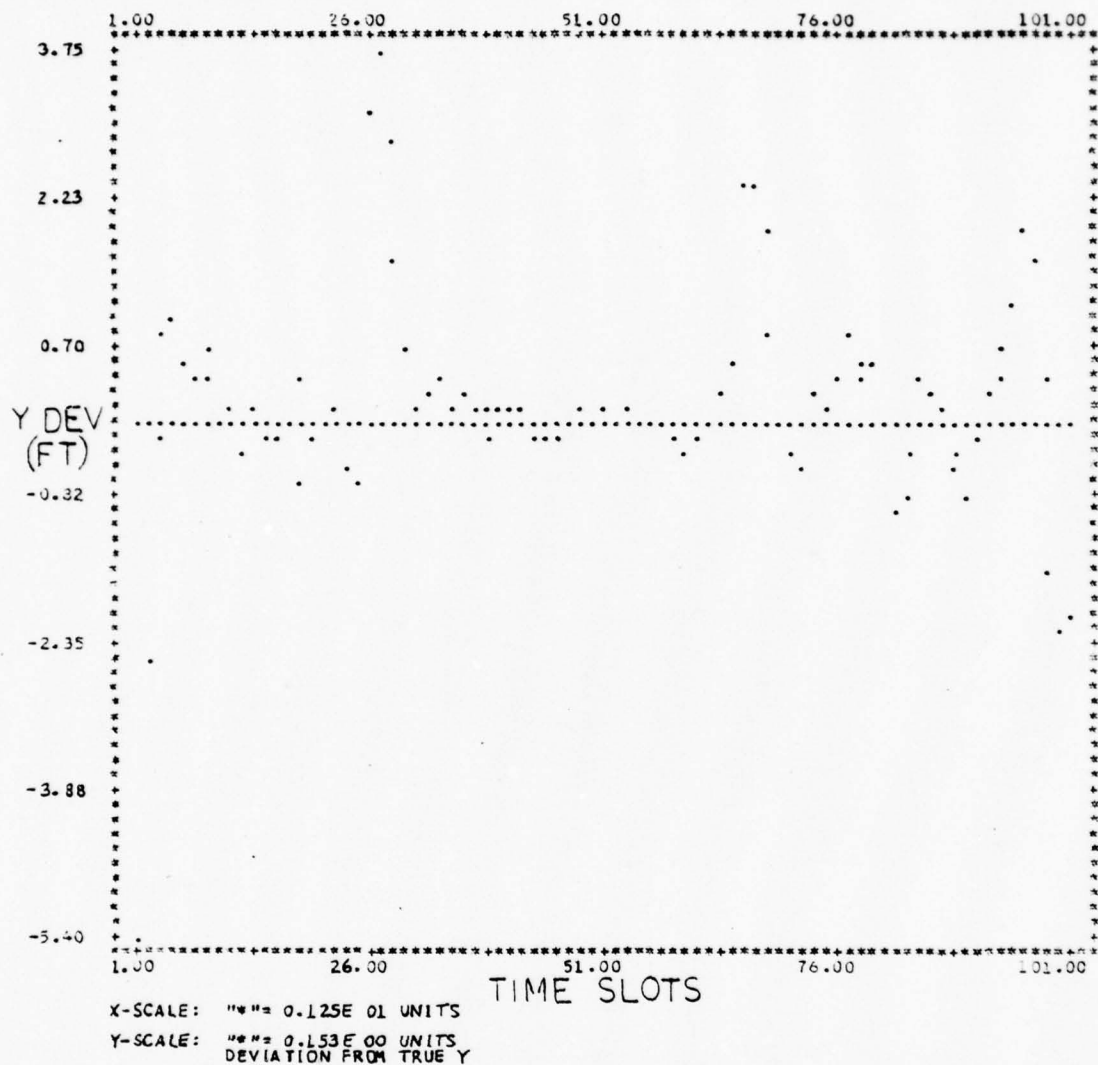


FIGURE 26: Y Deviation from True for a Track with 2/3-G and 1-G Turns with Noise in the Area of a Single Array -- Sequential Routine

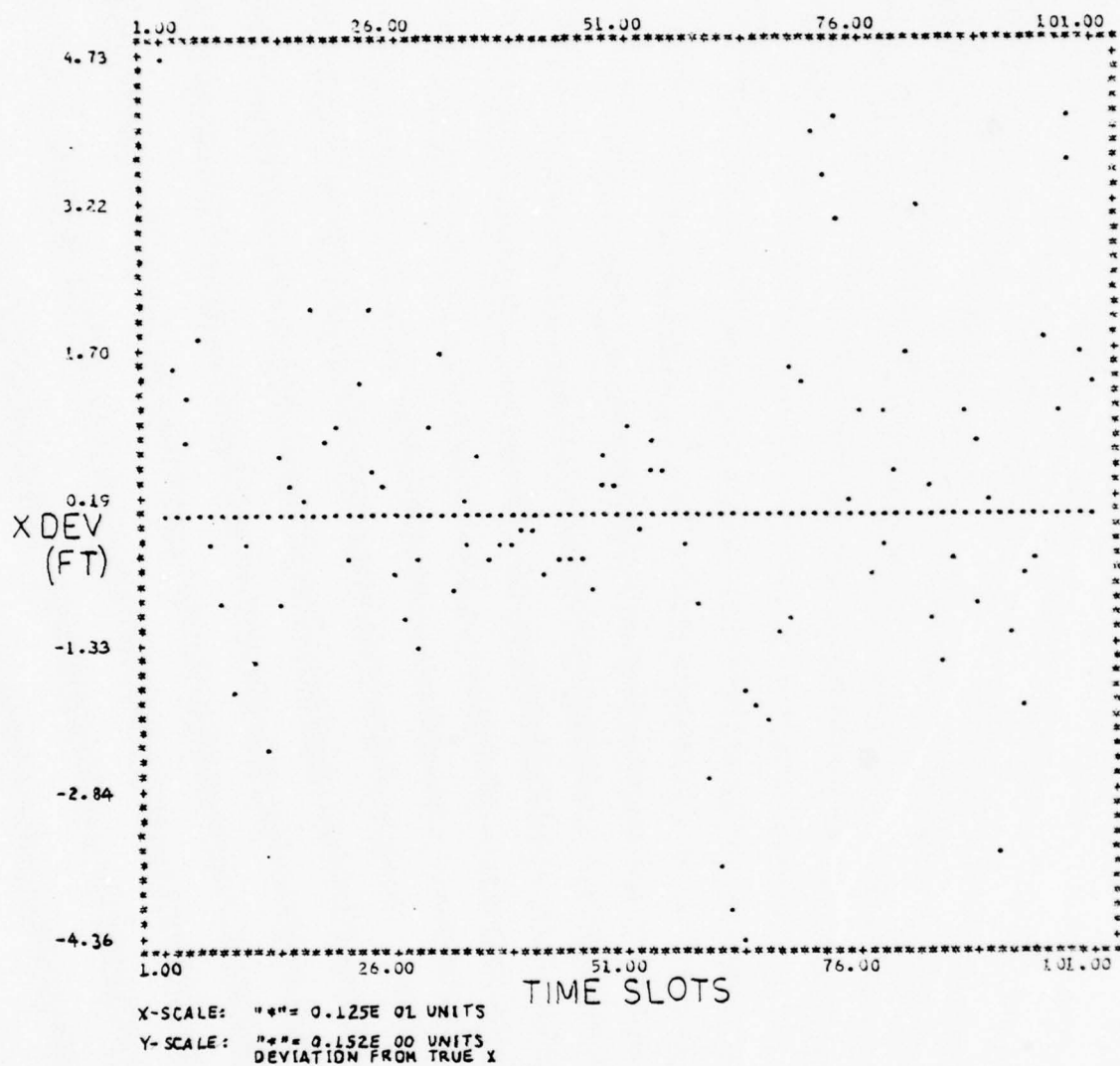


FIGURE 27: X Deviation from True for a Straight Running
42 Knot Target with Noise in the Area of a
Single Array -- Traditional Routine

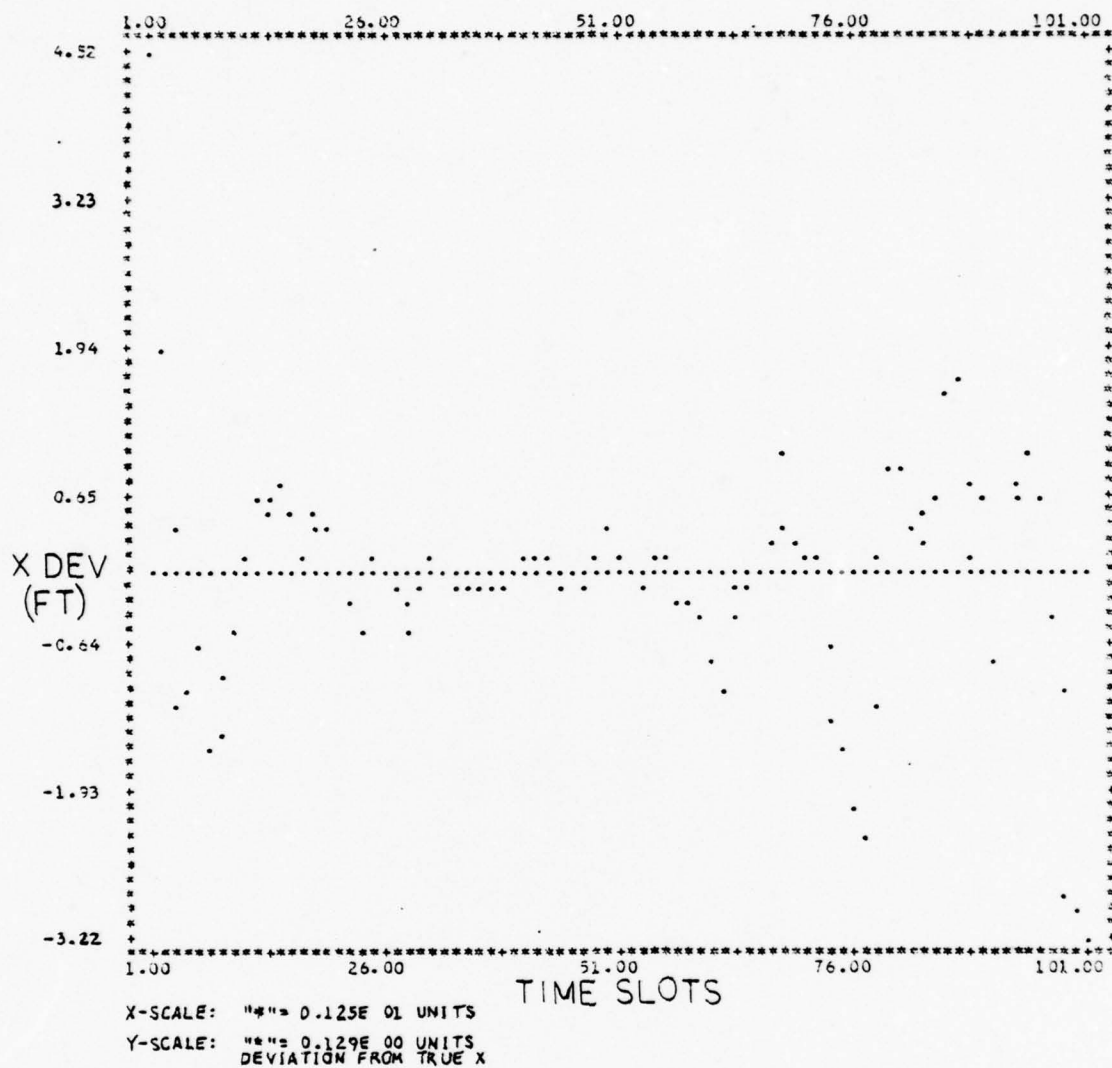


FIGURE 28: X Deviation from True for a Straight Running
42 Knot Target with Noise in the Area of a
Single Array -- Sequential Routine

Figures 29 and 30. Figure 29 depicts the estimate's deviation from true in the X direction for the traditional routine and Figure 30 for the sequential. In this track the target maneuvered, using $1/3$ -G turns, at time slots 25 and 65. A slightly better performance was also exhibited by the traditional routine in the Y direction.

In tests in which more radical maneuvers were made, the traditional routine again exhibited slightly better performance in noise. Figure 31, shows the deviation in the X direction for a track with a 1-G turn at time 25 and a $2/3$ -G turn at 65 using the sequential routine. Comparison with Figure 32, which is the same track using the traditional routine, indicates that the performances were similar with the traditional routine having a slight edge.

E. SERIES FIVE

In the last series of tests, the targets were tracked through multiple arrays using the handoff scheme described in the previous section. Figure 33 is a geographic plot of a typical track including hydrophone positions. Figure 34 depicts the estimate's deviation from true in the X direction using the traditional routine for a straight running target in noise. Figure 35 is the X deviation for the same track using the sequential routine. Through the multiple arrays, both routines were comparable, and the handoff from array to array was smooth with no glitches. Figures 36 and 37 also show comparable results for deviation in the Y direction for this same

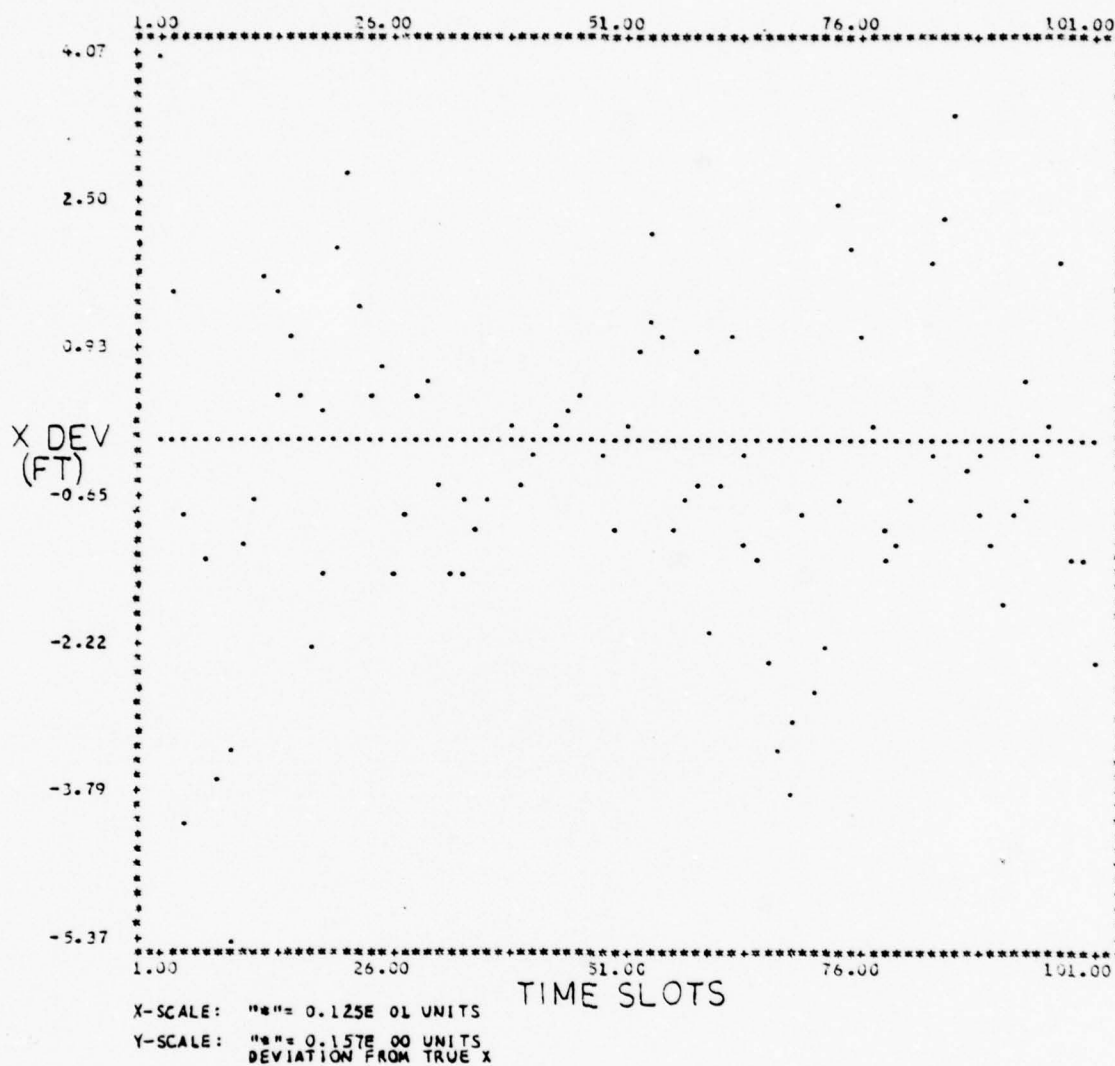


FIGURE 29: X Deviation from True for a Track with 1/3-G Turns at 42 Knots with Noise in the Area of a Single Array -- Traditional Routine

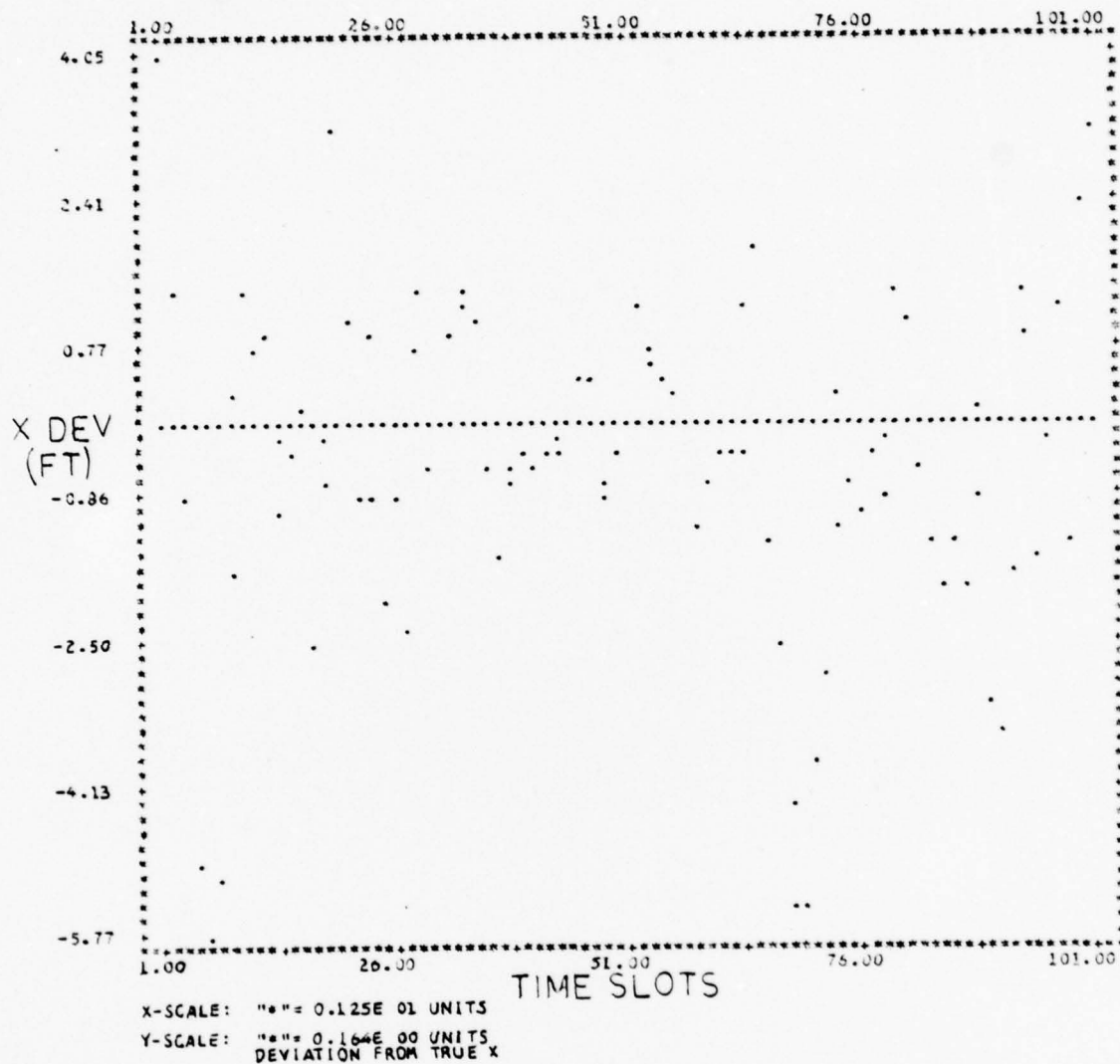


FIGURE 30: X Deviation from True for a Track with 1/3-G Turns at 42 Knots with Noise in the Area of a Single Array -- Sequential Routine

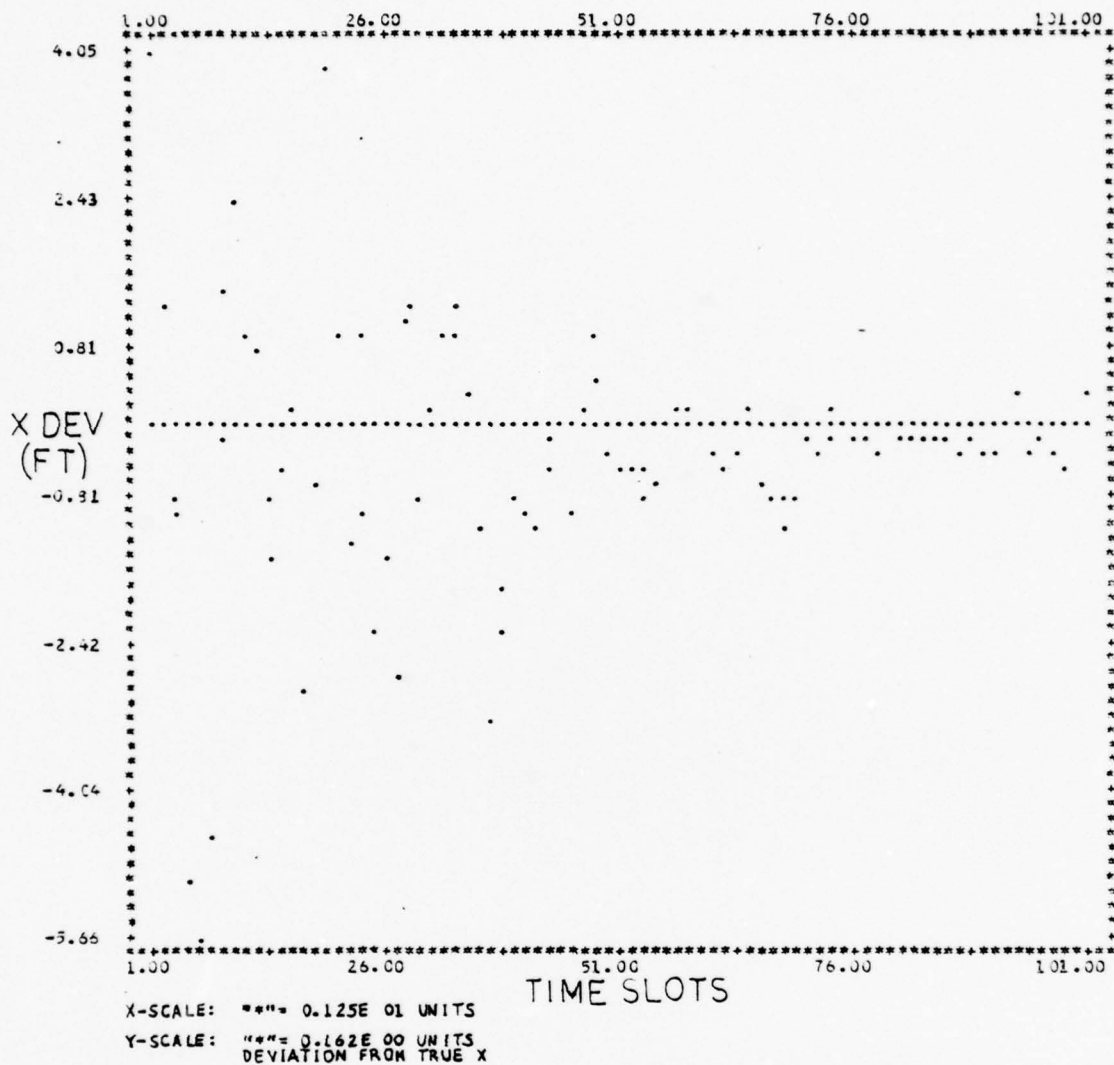


FIGURE 31: X Deviation from True for a Track with 2/3-G and 1-G Turns at 42 Knots with Noise in the Area of a Single Array -- Sequential Routine

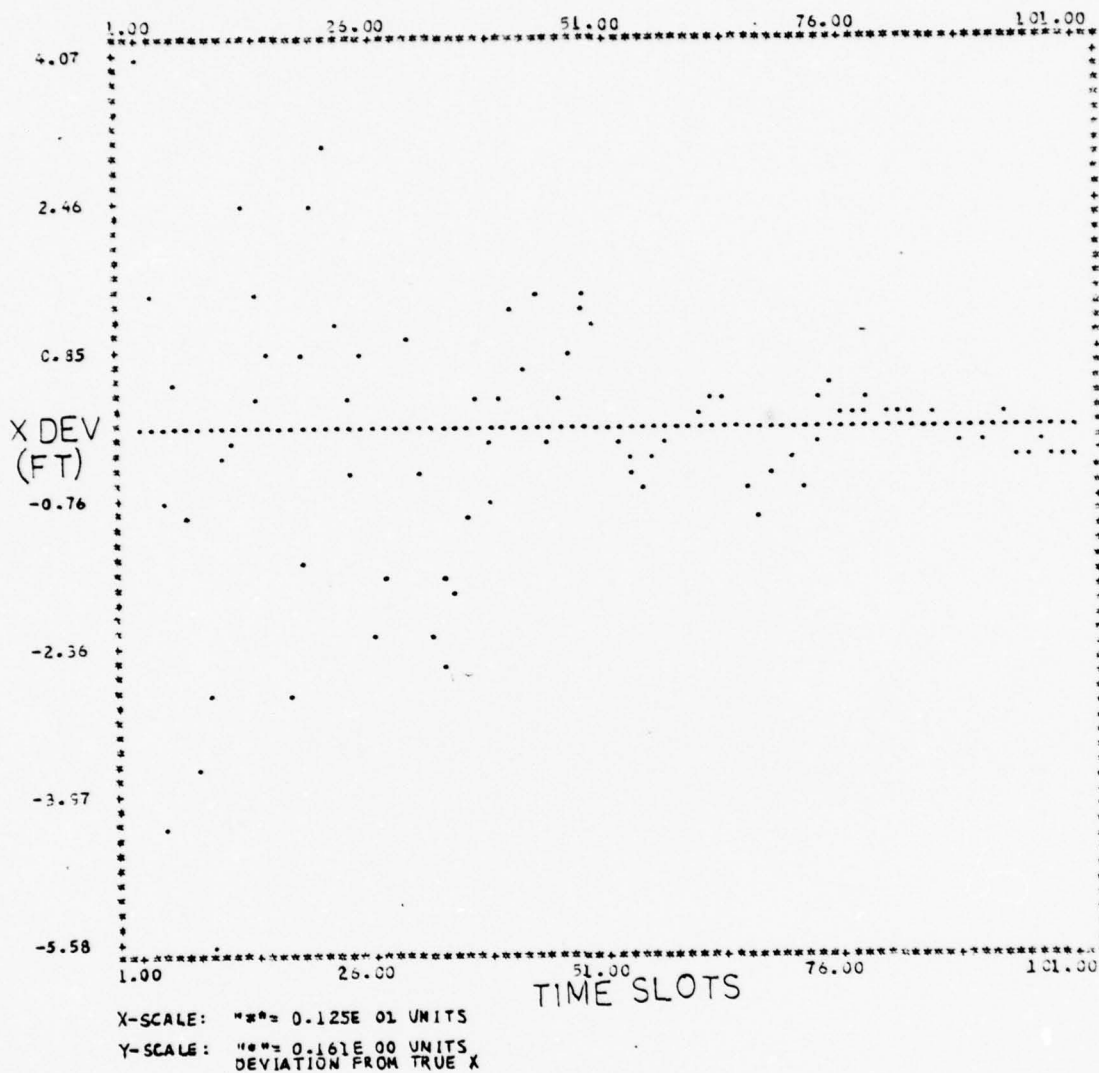


FIGURE 32: X Deviation from True for a Track with 2/3-G
and 1-G Turns at 42 Knots with Noise in the
Area of a Single Array -- Traditional Routine

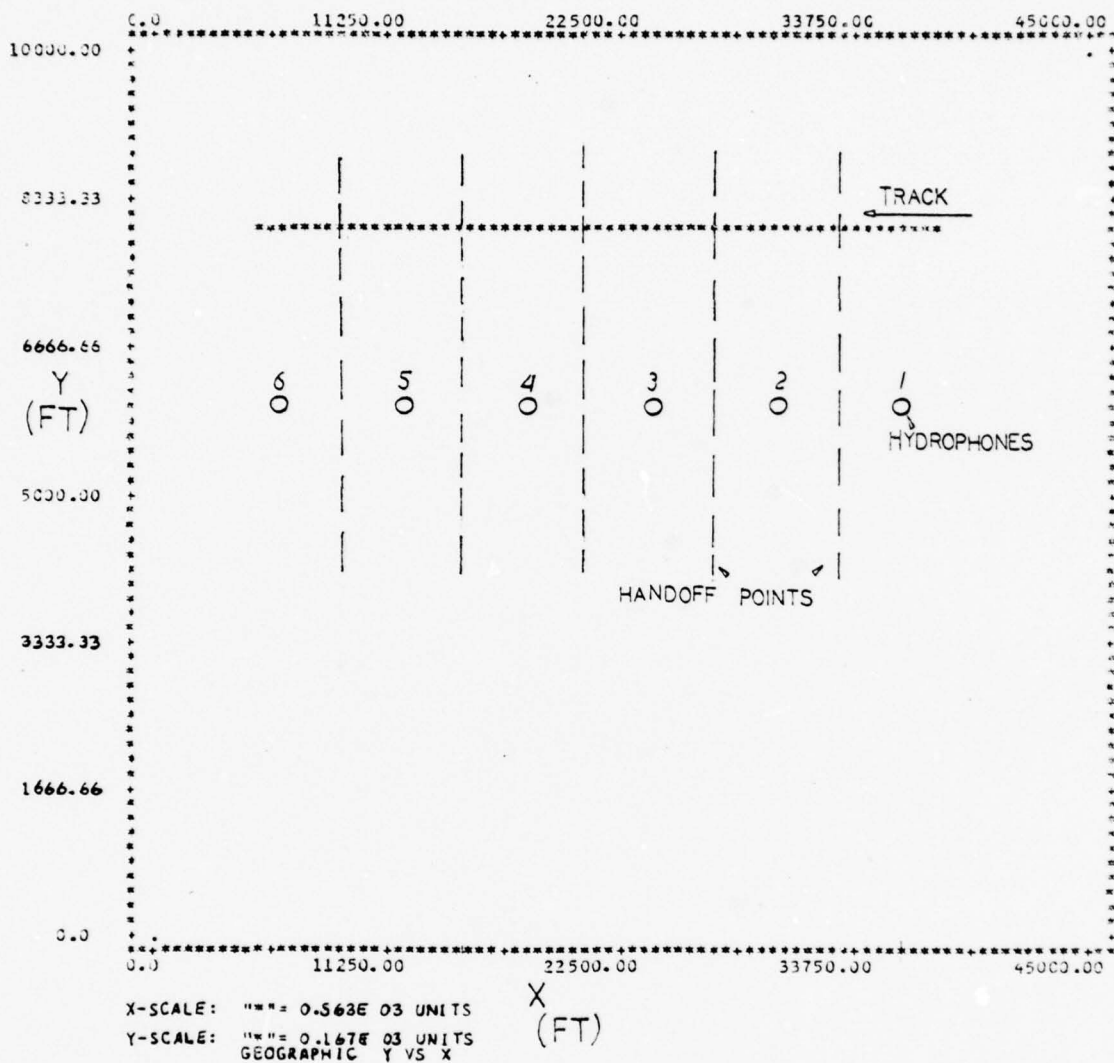


FIGURE 33: Geographic Plot of a Straight Running Track through Multiple Arrays

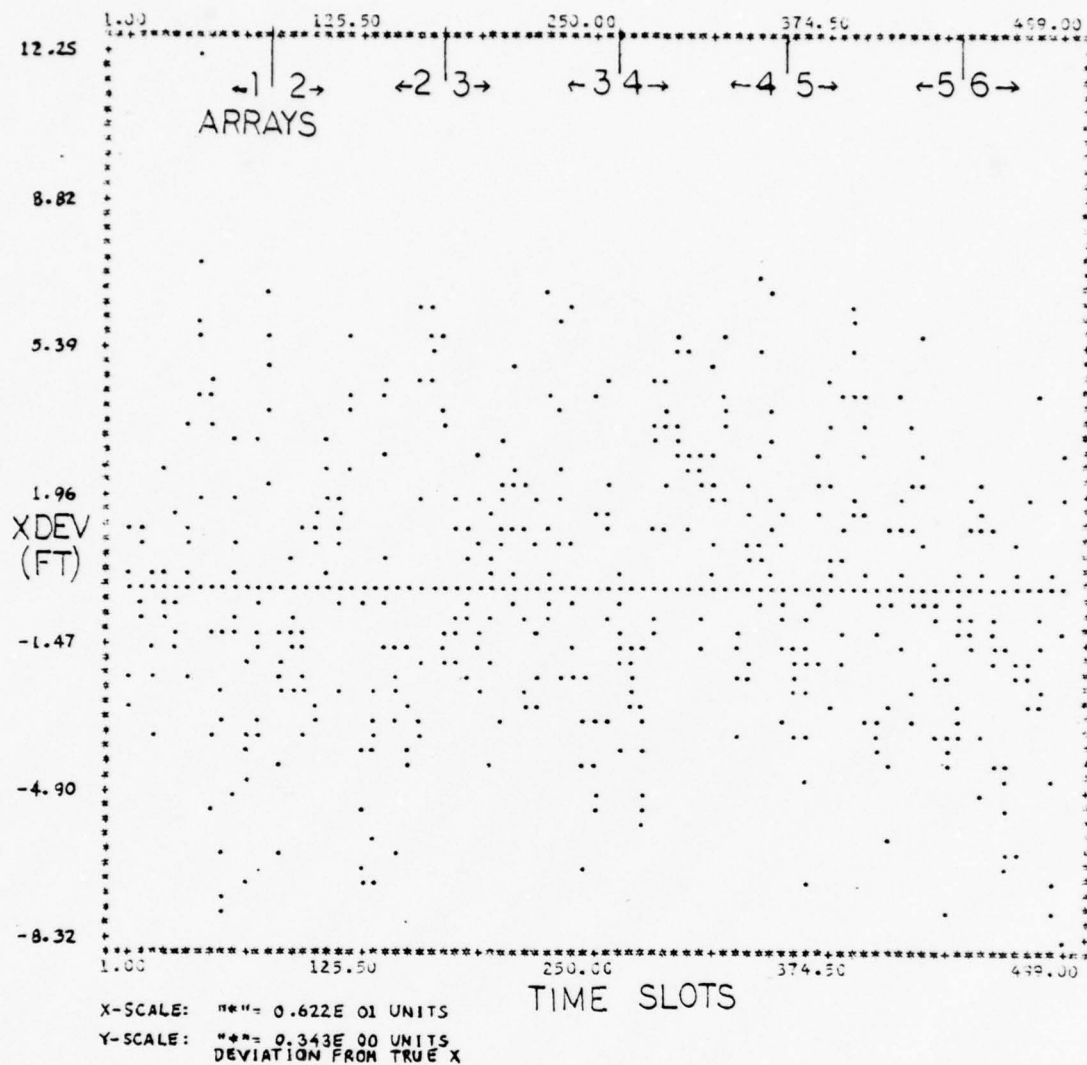


FIGURE 34: X Deviation from True for a Straight Running Track through Multiple Arrays with Noise -- Traditional Routine

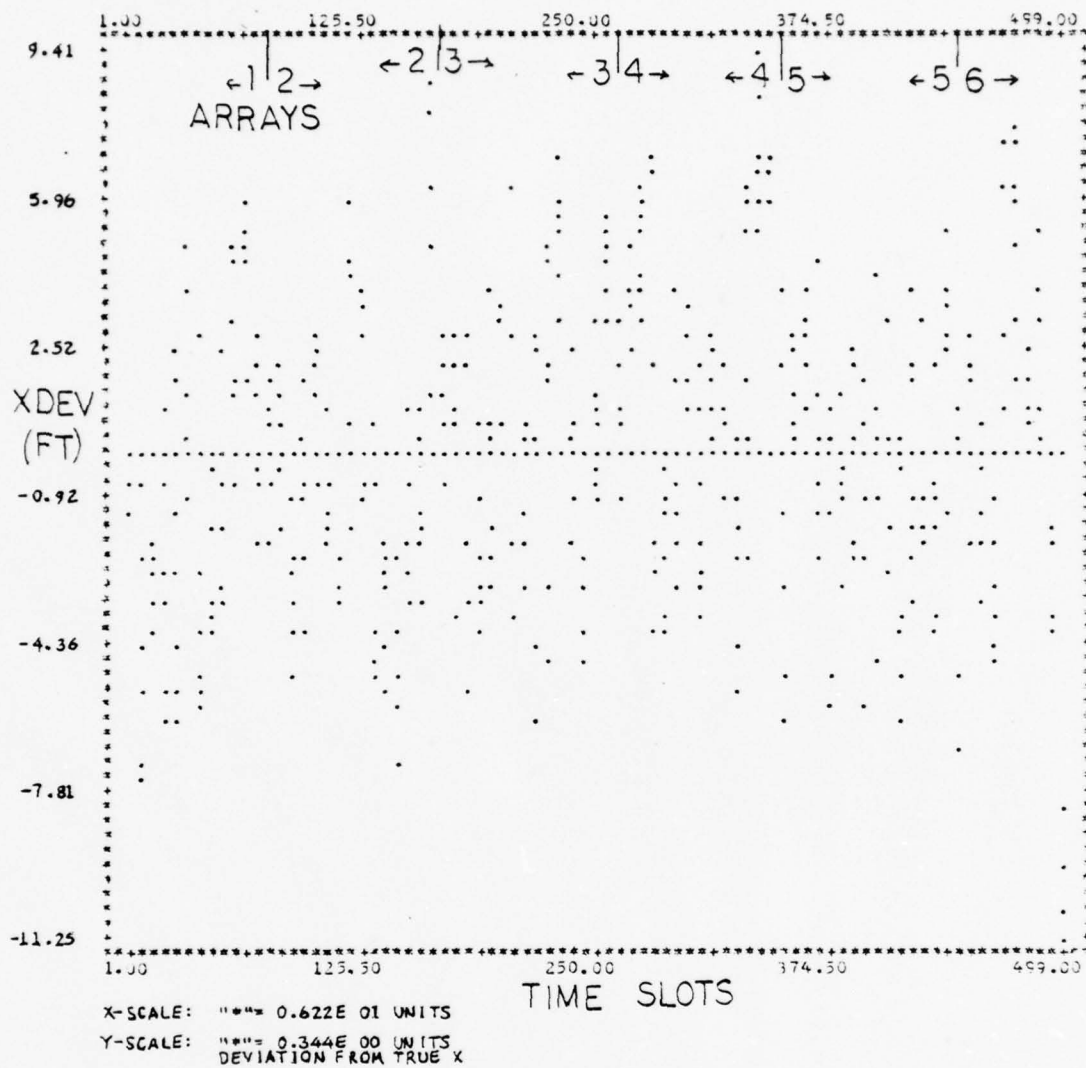


FIGURE 35: X Deviation from True for a Straight Running
Track through Multiple Arrays with Noise --
Sequential Routine

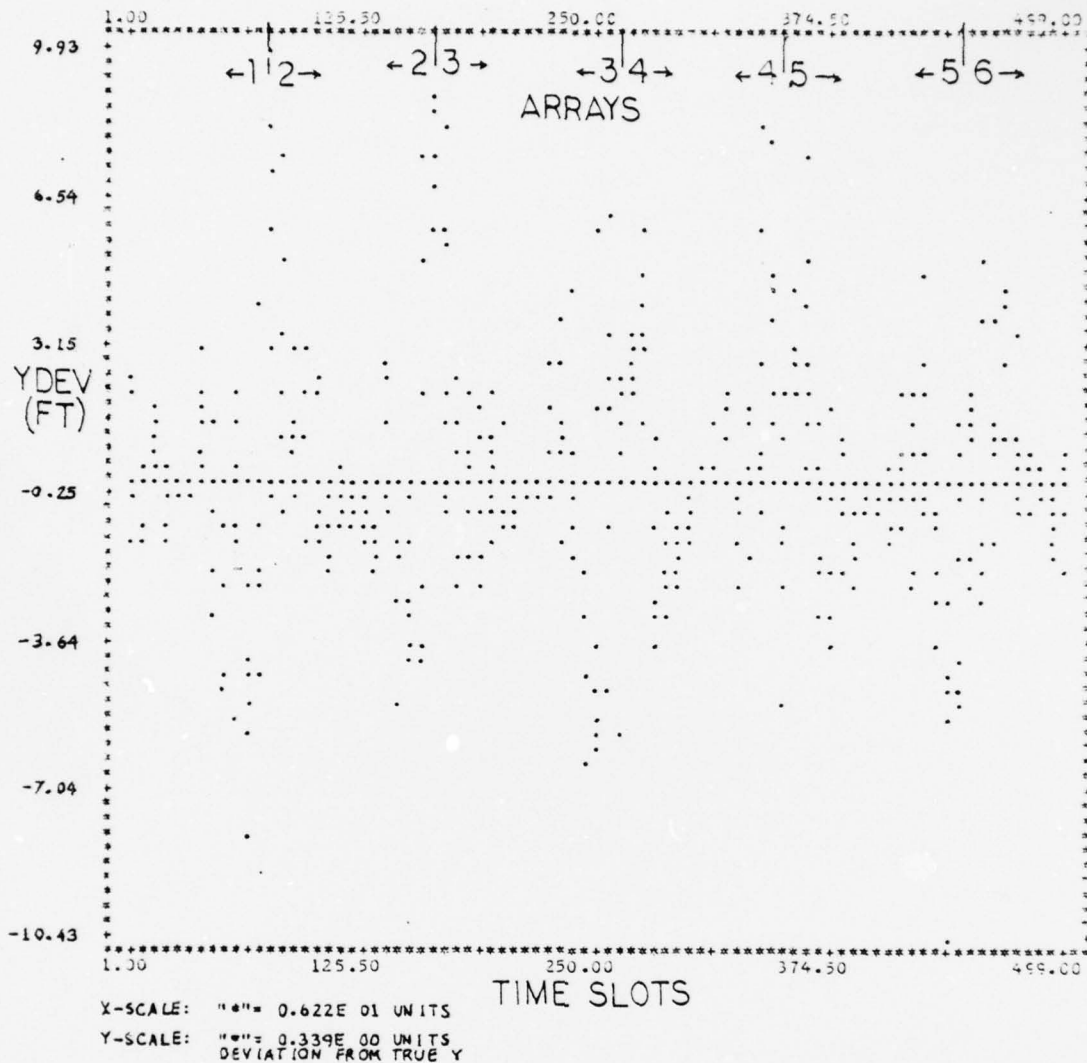


FIGURE 36: Y Deviation from True for a Straight Running
Track through Multiple Arrays with Noise --
Traditional Routine

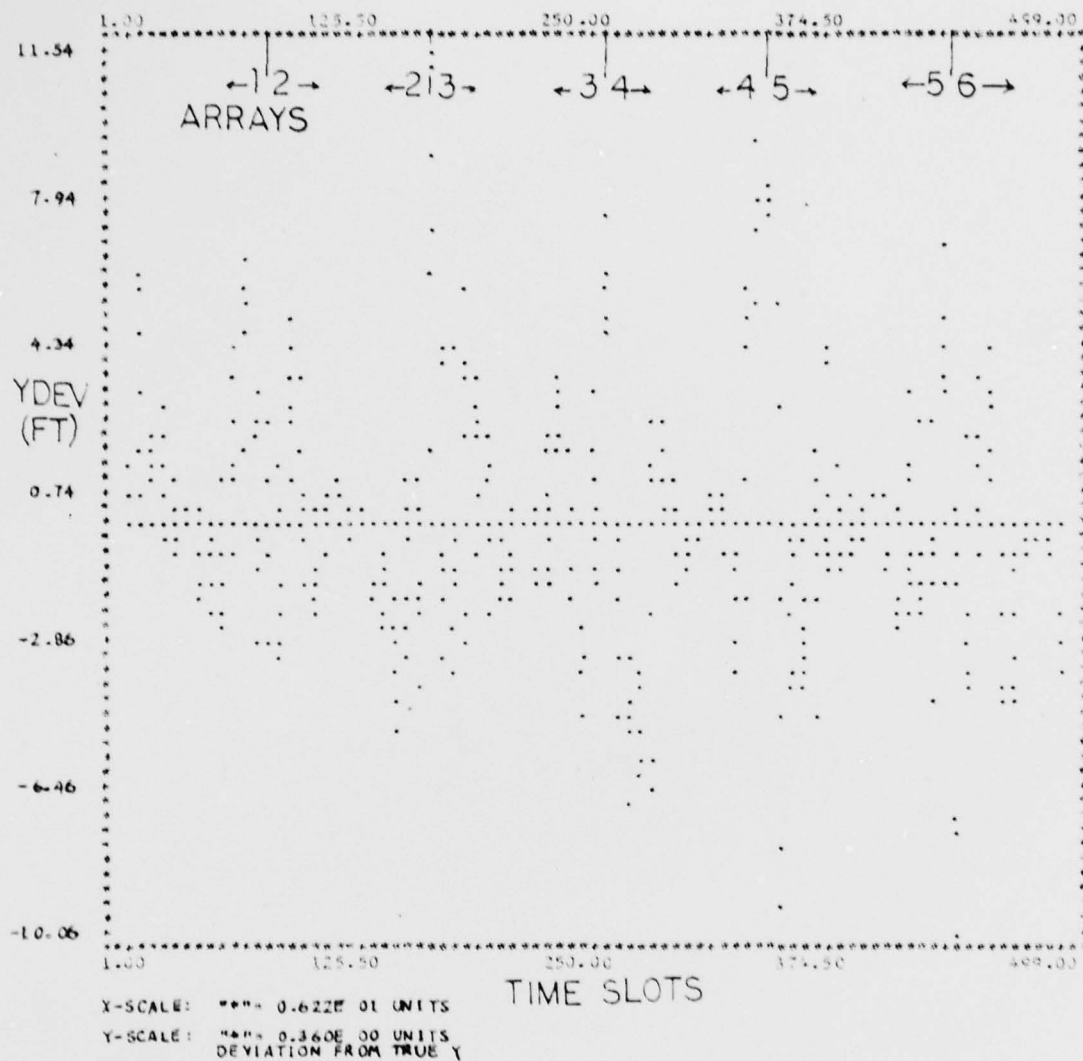


FIGURE 37: Y Deviation from True for a Straight Running Track through Multiple Arrays with Noise -- Sequential Routine

run. The target was traveling at 40 knots and handoff was accomplished at the following times:

| | | |
|----------------|-----|---------------|
| Handoff 1 to 2 | --- | Time Slot 77 |
| Handoff 2 to 3 | --- | Time Slot 168 |
| Handoff 3 to 4 | --- | Time Slot 260 |
| Handoff 4 to 5 | --- | Time Slot 352 |
| Handoff 5 to 6 | --- | Time Slot 443 |

VII. CONCLUSIONS

Both extended Kalman filter routines designed will provide on-line, real time estimates of targets with various maneuvers up to 1-G turns.

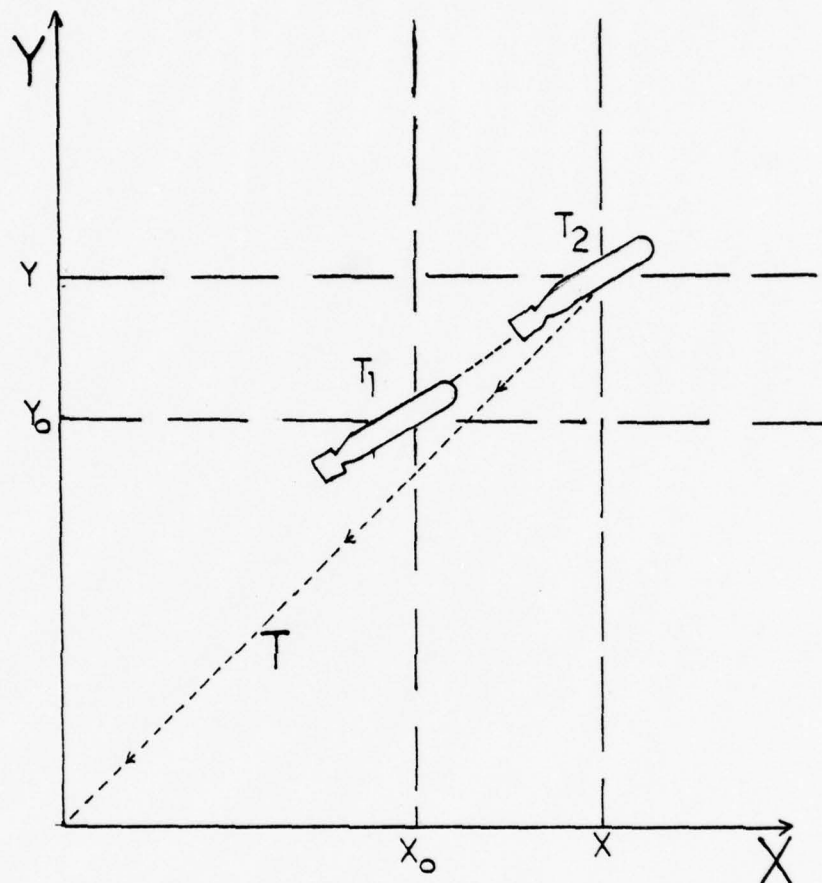
Implementation at the range computer facilities can be accomplished by loading the received transit times into a file in memory and reading them into the routine with subroutine DREAD. The hydrophone positions can also be read into a file to be referenced during operation. The initial covariance of estimation error $P(1/0)$, read in as a constant matrix, can be varied with the uncertainty of the targets initial positions and velocities. Any greatly erroneous time measurements caused by multipath signals or from noise spikes will cause the gains to zero, making the estimated position equal to the predicted, thus putting the filter in 'coast', and preventing catastrophic failure.

The system was remodeled to include the X and Y velocities in the linearizing matrix H. This model is depicted in Figure 38. With a constant Z, the targets position at time t_1 is $X_0 Y_0$. By the time the transmitted signal has reached the hydrophone array, the target with X velocity (V_X) and Y velocity (V_Y) has moved to the position at t_2 :

$$X = X_0 + V_X T$$

$$Y = Y_0 + V_Y T$$

where T is the transit time of the signal



MOVING TARGET in the X-Y PLANE

$$X = X_0 + V_x T$$

$$Y = Y_0 + V_y T$$

FIGURE 38: Geometry used to remodel the System
to include X and Y Velocities in the
H Matrix

$$T = 1/VEL \sqrt{X_0^2 + Y_0^2}$$

and VEL is the velocity of propagation of sound in water.

Including the new positions corresponding to the target location when the signal is received gives a new equation for the transit time.

$$T = \frac{1}{VEL} \sqrt{(X-V_X T)^2 + (Y-V_Y T)^2}$$

solving for T produces:

$$T = \frac{(XV_X + YV_Y) - \sqrt{(XV_X + YV_Y)^2 - (V_X^2 + V_Y^2 - VEL)(X^2 + Y^2 + Z^2)}}{(V_X^2 + V_Y^2 - VEL)}$$

Upon testing, this added complexity did not give a corresponding increase in the quality of the estimate.

In general, both routines were comparable with the sequential filter having the following advantages:

1. Less deviation and quicker lock-on time after a modest maneuver
2. With the three-sigma gate utilized a noise spike encountered will negate data from only that particular hydrophone
3. A matrix inversion is not necessary.

APPENDIX A

PROGRAM DESCRIPTION AND FEATURES

Two programs were written to implement the extended Kalman filter routine for torpedo tracking. The first, THEFIV, utilizes the sequential approach described in section IV-A, and the second a traditional matrix inversion approach as described in section IV-B. Both routines use the same utility programs and are modularized for ease of implementation.

1. THEFIV - Sequential Routine

This program is general in nature and many of the parameters of the Kalman routine are variable including:

- a. The number of states in the routine - N
- b. The number of random forcing functions - M
- c. The number of measurements - J
- d. Data rate or sample time - T_0
and $\phi(1,2)$, $\phi(3,4)$
- e. Number of time slots - JTIME

The constant matrices PHI, R, COVW and GAMMA are read in using subroutines in the utility program AUX. The filter is initialized with $\underline{P}(1/0)$ and $\underline{X}(1/0)$ (initial covariances of estimation error and states) also using AUX. The first state estimate is at time 1 and continues until $ITIME = JTIME+1$. True measurement times (ZI) are read in, four for each time slot (T_C, T_X, T_Y, T_Z), using the subroutine DREAD listed in AUX and corrupted by zero-mean, white Gaussian noise using the

IBM-360 subroutine SNORM. For each of the four time measurements the corresponding row of the linearizing H matrix is calculated in the utility subroutine CHROW, and the corresponding gain matrix column GI is found. These row and column values are then utilized in forming the covariance of estimation error for the particular time measurement PI. Next the estimate of the observation time $\hat{M}(X)$ from that particular hydrophone called ZHAT is formed using the subroutine CZHAT and the residual ZDIFF = ZI-ZHAT. Finally, the estimate of the states XI based on one time measurement is calculated, and the process is repeated for the next measurement. After four iterations, XI becomes the state estimate and PI becomes the updated covariance of estimation error PKK, and the predictions of the states and covariances XKKM1 and PKKM1 are formed.

For testing, this program used the IBM-360 library subroutine PLOTP to obtain plots of the states and covariances versus time, estimate deviation from true versus time and a geographic track.

2. THESIS - Traditional Routine

THESIS utilized the same format as THEFIV. The parameters N,M,JS, sample time and number of time slots were still variable, and the filter initial conditions and constant matrices were read in by the subroutines located in AUX. True time measurements were again read in for each time slot through DREAD and corrupted with noise from SNORM. The linearizing H matrix was formed row by row using subroutine CHROW and used

to calculate the gain matrix G . The symmetric matrix inversion in the gain equation was done by the IBM-360 library subroutine SINV. The estimate vector of the observations \underline{ZHAT} is formed iteratively by CZHAT and used to calculate the residual vector \underline{ZDIFF} . Next the estimate XKK and updated covariance of error matrix PKK is calculated once for each time slot. Finally, the predictions $XKKM1$ and $PKKM1$ are formed before the process is repeated for the next time slot. PLOTP was again utilized for the output graphs.

3. Utility Programs

These subroutines were designed to be used for repetitive calculations and processes. The first, AUX performs all data input functions and matrix manipulations including:

- a. PROD - multiplying two matrices
- b. MMULT - multiplying a matrix and a vector
- c. VMULT - multiplying two vectors
- d. MREAD - reading in a matrix
- e. TRANS - transposing a matrix
- f. ADD - adding two matrices
- g. VREAD - reading in a vector
- h. DREAD - reads in a matrix containing observed time data
- i. TRREAD - reads in a matrix containing true positional data for comparison

The second utility subroutine CZHAT calculates the estimate of the observation times (T_C, T_X, T_Y or T_Z) using the predicted state values ($\hat{X}(K/K-1)$).

The subroutine CHROW calculates a row of the linearizing H matrix, each row corresponding to a particular observation time measurement.

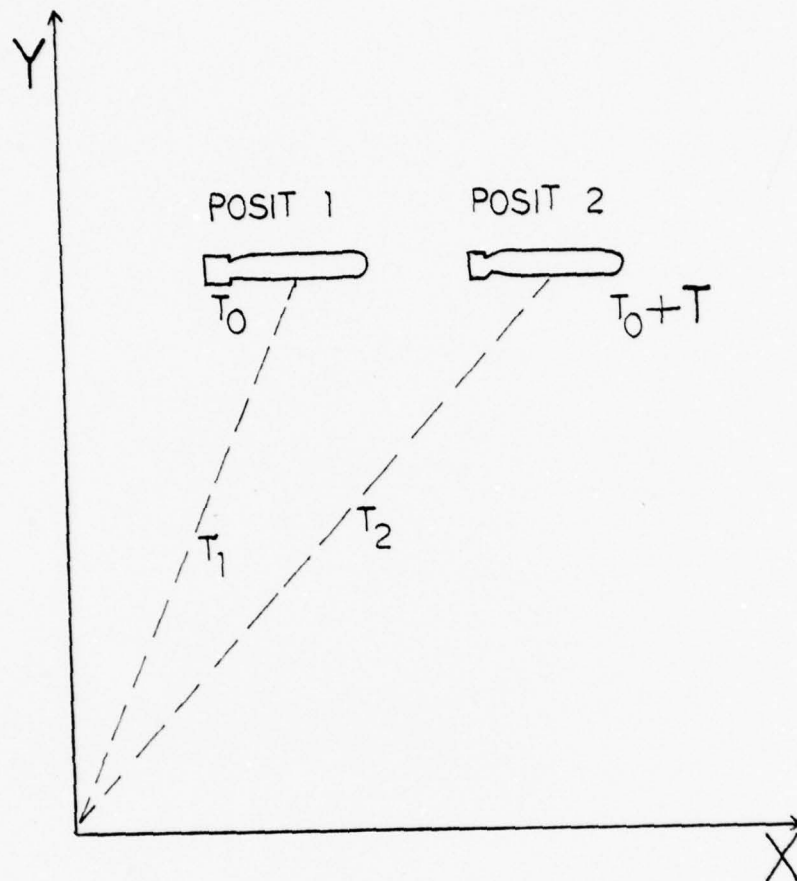
A. VARYING STATE TRANSITION MATRIX

In the $1/S^2$ model that represents the dynamics of the Kalman filter, the State Transition and GAMMA matrices are as follows:

$$\Phi = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & T & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\Gamma = \begin{bmatrix} T^2/2 & 0 & 0 \\ T & 0 & 0 \\ 0 & T^2/2 & 0 \\ 0 & T & 0 \\ 0 & 0 & T \end{bmatrix}$$

where T is the sampling time. In the tracking problem, this time is not always constant. As is shown in Figure 39 it varies with the transit times required from positions in adjacent time slots. In Figure 39, the target is at position 1 at arbitrary time t_0 , and is at position 2, the next time slot, at t_0+T where T is the time between pings of the target transducer = 1.31 seconds. The times that the transmitted



Moving Target in the X-Y Plane

$$T_s = (t_0 + T + t_2) - (t_0 + t_1)$$

$$T_s = T + (t_2 - t_1)$$

FIGURE 39: Geometry used for the Time Varying State Transition Matrix

signals are received by the hydrophone array are:

Time signal received from position 1 = $t_0 + t_1$

Time signal received from position 2 = $t_0 + T + t_2$

where t_1 and t_2 are the transit times from position 1 and position 2 respectively. The difference between these two times signals are received is the sampling time T_S .

$$T_S = (t_0 + T + t_2) - (t_0 + t_1)$$

$$T_S = T + (t_2 - t_1)$$

Therefore, the sampling time differs each time slot from the clocked ping time by the difference in the adjacent slot transit times.

In the programs THEFIV and THESIS, this difference is calculated using the average measured times from the four hydrophones for each time slot.

B. ADAPTIVE Q

The Q matrix which appears in the predicted covariance of error equation

$$\tilde{P}(K+1/K) = \tilde{\Phi} \tilde{P}(K/K) \tilde{\Phi}^T + \tilde{Q}(K)$$

and is formed by

$$\tilde{Q}(K) = \tilde{\Gamma} \text{COVW} \tilde{\Gamma}^T$$

is a measure of the amount of target maneuverability that can be handled by the filter. If more random excitations (or accelerations) by the target is expected, Q is increased which

in turn increases the covariances of estimation error $\tilde{P}(K+1/K)$ and the filter gains G . Corresponding the filter puts more emphasis on the incoming data and is better able to see and react to target turns. However, if the filter gains are increased the filter bandwidth is widened, which lets in more noise, and makes the filter more susceptible to error. The adaptive Q routine (Reference 4) varies the Q matrix as the velocities in the X and Y directions are increased or decreased. This routine was implemented in the subroutine QFIND with inputs of:

SIGACC = expected maximum acceleration in the X or Y
direction in ft/sec^2

SIGDIV = expected maximum acceleration in the Z direction
in ft/sec^2

SIGCC = expected maximum target course change in degrees/
sec

QFIND is listed in Appendix D.

APPENDIX B
SEQUENTIAL EXTENDED KALMAN FILTER
THEFIV

THE00010
THE00020
THE00030
THE00040
THE00050
THE00060
THE00070
THE00080
THE00090
THE00100
THE00110
THE00120
THE00130
THE00140
THE00150
THE00160
THE00170
THE00180
THE00190
THE00200
THE00210
THE00240

THE00260
THE00270
THE00280

THE00300
THE00310
THE00320

THE00340
THE00350
THE00360
THE00370
THE00380
THE00390
THE00400
THE00610
THE00620
THE00630
THE00640
THE00650
THE00660
THE00670
THE00680
THE00690
THE00700

```

COMMON XKKM1(5), PKKM1(5,5), PKK(5,5), XKK(5)
DIMENSION PHI(5,5), GAMMA(5,3), R(4,4), PHIPKK(5,5), PHIT(5,5),
IPKTEMP(5,5), HROW(5), PI(5,5), PDUM(5,3), QTEMP(5,5), XI(510),
IGNUM(5), GI(5), XI(5), ZI(4), GAMMAT(3,5), Q(5,5), COVM(3,3),
IX5(510), KOUNT(510), AKOUNT(510), PVEC(510), PI(510,5), X3(510),
ITRUX(510), TRUV(510), TRUZ(510), ID(3), CI(2), C2(2), C3(2),
IC4(2), C5(2), C6(2), C7(2), XP(5,510), XDIFF1(510), XDIFF3(510),
XDIFF5(510), ZC(4), HYDRO(6,12), XB(4), YB(4), ZE(4)
N=5
M=3
JS=4
I8=1
TO=1.31 200) JTIME
READ(5,13)
FORMAT(13)
200 READ(5,201) SIGACC, SIGDIV, SIGCC
FORMAT(3F4.2)
201 SIGCC=SIGCC*3.14159/180.
DO 799 I=1,6
799 READ(7,739) (HYDRO(I, J), J=1,12)
739 FORMAT(12F6.1)

C LOAD X(0/-1)
CALL VREAD(XKKM1, N)

C LOAD F(0/-1)
CALL MREAD(PKKM1, N, N)

C READ IN CONSTANT MATRICES-GET TRANSPOSES

CALL MREAD(PHI, N, N)
CALL MREAD(R, JS, JS)
CALL MREAD(COVM, M, M)
CALL MREAD(GAMMA, N, M)
CALL TRANS(GAMMA, N, M, GAMMAT)
CALL TRANS(PHI, N, N, PHIT)

C START THE TIME SLOT LOOP
C
ITIME=JTIME+1
XT=3600.
SW=XT-3000.
DO 99 KK=1, ITIME
DC 600 I3=1, 4
XB(I3)=HYDRC(I8, 3*I3-2)
YB(I3)=HYDRO(I8, 3*I3-1)

```

THEFIV (con't)

```

600 ZB(I3)=HYDRC(I8,3*I3)
    IF(XKKM1(1).GT.SW)GO TO 610
    I8=I8+1
750 WRITE(6,759)I8,KK
    FORMAT(1X,'ARRAY',2X,I2,'STARTS TRACKING AT TIME',2X,I3)
    SW=SW-6000.
    XT=XT-6000.

C COMPUTE THE TRUE TIMES
C
610 CALL DREAD(ZI,JS)
C TIME VARY THE STATE TRANSITION MATRIX PHI
C
    T2=(.25)*(ZI(1)+ZI(2)+ZI(3)+ZI(4))
    IF(KK.EQ.1)T1=T2
    PHI(1,2)=TO-(T2-T1)
    PHI(3,4)=PHI(1,2)
    PHIT(2,1)=PHI(1,2)
    PHIT(4,3)=PHI(1,2)
    T1=T2
    A14=PHI(1,2)

    CALL TRREAD(TD,M)
    TRUX(KK)=TD(1)
    TRUY(KK)=TD(2)
    TRUZ(KK)=TD(3)

C FIRST GET HROW-CALCULATE GAIN,ESTIMATE,COVARIANCE OF
C ERROR BASED ON ONE TIME MEASUREMENT-TC,TX,TY OR TZ
711 DO 97 I=1,JS
    CALL CHROW3(I,HROW,XB,YB,ZB)
    CALL MMULT(PKKM,HROW,N,N,GNUM)
    CALL VMULT(HROW,GNUM,N,GTEMP)
    GDENOM=GTEMP+R(I,I)
    DO 16 IX=1,N
    GI(IX)=GNUM(IX)/GDENOM

16 C THIS IS THE FIRST GAIN COLUMN
C CALCULATE THE COVARIANCE OF ERROR - PI
    DO 77 IP=1,N
    DO 79 JP=1,N
    PDUM(IP,JP)=(-1.*GI(IP))*HROW(JP)
    IF(IP.EQ.JP)PDUM(IP,JP)=1.+PDUM(IP,JP)
79 CONTINUE
77 CONTINUE

```

```

THE00710
THE00720
THE00730
THE00740
THE00750
THE00760
THE00770
THE00880
THE00890
THE00900
THE00910
THE00920
THE00930
THE00940
THE00950
THE00960
THE00970
THE00980
THE00990
THE01000
THE01010
THE01020

THE01030
THE01040
THE01050
THE01060
THE01070

THE01100
THE01110
THE01120
THE01130
THE01140
THE01150
THE01160
THE01170
THE01180

THE01210
THE01220
THE01230
THE01240
THE01250
THE01260
THE01270

```


THE01280
THE01340
THE01360
THE01370
THE01430

THE01460
THE01470
THE01480
THE01490
THE01500
THE01510

THE01540
THE01560
THE01570
THE01580
THE01590
THE01600
THE01610

THE01630
THE01640
THE01650
THE01880
THE01890
THE01900
THE01910
THE01920
THE01930
THE01940
THE01950
THE01980
THE01990

THE02040
THE02050
THE02060
THE02070
THE02080
THE02090
THE02200

THEFIV (con't)

CALL PROD (PDUM,PKKM1,N,N,N,PI)

C CALCULATE THE FIRST MEASUREMENT PREDICTION

122 CALL CZHAT3(I,ZHAT,XB,YB,ZB)
ZDIFF=ZIC-ZHAT

C SET

THE GATE
PK1=ABS(PI(1,1))
PK2=ABS(PI(3,3))
PK3=ABS(PI(5,5))
IF((PK1.GE.PK3).AND.(PK1.GE.PK5))P=PK1
IF((PK3.GE.PK1).AND.(PK3.GE.PK5))P=PK3
IF((PK5.GE.PK1).AND.(PK5.GE.PK3))P=PK5
PGATE=P/4860
RGATE=SQRT(ABS(R(1,1)))
GATE=3.*SQRT(PGATE*RGATE)
IF(ZDIFF.LT.GATE)GO TO 500
WRITE(6,501)KK
501 FORMAT(10,'GATE HAS BEEN EXCEEDED TIME',I4)
502 DO 502 LG=1,N
GI(LG)=0.0

C CALCULATE THE ESTIMATE BASED ON ONE MEASUREMENT

500 DO 17 IZ=1,N
17 XI(IZ)=XKKM1(IZ)+GI(IZ)*ZDIFF

12 IF(I.EQ.4)GO TO 56
19 DO 19 IQ=1,N
XKKM1(IQ)=XI(IQ)
DO 23 IQ=1,N
DO 18 JQ=1,N
PKKM1(IQ,JQ)=PI(IQ,JQ)
CONTINUE
CONTINUE

C ORIGINAL X(0/-1) WAS XKKM1, UPDATED AFTER 1
C MEASUREMENT AND CALLED IT XI, THEN XKKM1=XI, WENT
C THRU ITERATION AGAIN, AFTER XKKM1 UPDATED FOR EACH
C MEASUREMENT XKK=XI AND PKK=PI

56 DO 57 ID=1,N
XKK(ID)=XI(ID)
DO 58 JD=1,N
PKK(ID,JD)=PI(ID,JD)
CONTINUE

THEFIV (con't)

THE02210
THE02220
THE02230
THE02240

THE02260
THE02270
THE02280
THE02290
THE02300
THE02310
THE02320
THE02330
THE02340
THE02350
THE02360

THE02380
THE02390
THE02400
THE02410
THE02420
THE02430
THE02440
THE02450
THE02460
THE02470
THE02480
THE02490
THE02500
THE02510
THE02520
THE02530
THE02540
THE02550
THE02560
THE02570
THE02580
THE02590
THE02600
THE02610
THE02620
THE02630
THE02640
THE02650
THE02660
THE02670
THE02680
THE02690
THE02700
THE02710
THE02720
THE02730
THE02740
THE02750

```

XDIFF1(KK)=TRUX(KK)-XKK(1)
XDIFF3(KK)=TRUY(KK)-XKK(3)
XDIFF5(KK)=TRUZ(KK)-XKK(5)

C CALCULATE THE PREDICTIONS FOR PKKM1
      CALL QFIND(KK,SIGACC,SIGDIV,SIGCC,A14,Q)
      CALL PROD(PHI,PKK,N,N,V,PHIPKK)
      CALL PROD(PHIPKK,PHIT,N,N,N,PKTEMP)
      CALL ADD(PKTEMP,Q,N,N,PKKM1)

C CALCULATE THE PREDICTIONS FOR XKKM1
      CALL MMULT(PHI,XKK,N,N,XKKM1)
      DO 41 IG=1,N
        XP(IG,KK)=XKK(IG)
      DO 38 II=1,N
        DO 39 JJ=1,N
          P1(KK,II,JJ)=PKK(II,JJ)
        CONTINUE
      C1(1)=290.
      C1(2)=310.
      C3(1)=0.
      C3(2)=510.
      C4(1)=0.
      C4(2)=45000.
      C5(1)=0.
      C5(2)=10000.
      DO 91 KK=1,itime
        X1(KK)=XP(1,KK)
        X3(KK)=XP(3,KK)
        X5(KK)=XP(5,KK)
        DO 92 IK=1,itime
          KOUNT(IK)=IK
          DO 113 IF=1,itime
            AKOUNT(IF)=FLOAT(KOUNT(IF))
          DO 238 II=1,N
            JJ=II
            DO 240 KK=1,itime
              PVEC(KK)=P1(KK,II,JJ)
              WRITE(6,900)
            CALL PLOTP(AKOUNT,PVEC,itime,0)
            WRITE(6,800)II,JJ
            FORMAT(23X,'PKK('',II'',',',II'',',',',VS TIME')
            FORMAT(1,')
            WRITE(6,900)
          CALL PLOTP(C3,C4,2,1)

```

THEFIV (con't)

THE02760
THE02770
THE02780
THE02790
THE02800
THE02810
THE02820
THE02830
THE02840
THE02850
THE02860
THE02870
THE02880
THE02890
THE02900
THE02910
THE02920
THE02930
THE02940
THE02950
THE02960
THE02970
THE02980
THE02990
THE03000
THE03010
THE03020
THE03030
THE03040
THE03050
THE03060
THE03070

```

      CALL PLOTP(AKOUNT,X1,ITIME,2)
      CALL PLOTP(AKOUNT,TRUX,ITIME,3)
      WRITE(6,801)
      FORMAT(25X,'XPOSIT VS TIME')
      WRITE(6,900)
      CALL PLOTP(AKOUNT,XDIFF1,ITIME,0)
      WRITE(6,805)
      FORMAT(25X,'DEVIATION FROM TRUE X')
      WRITE(6,900)
      CALL PLOTP(C3,C5,2,1)
      CALL PLOTP(AKOUNT,X3,ITIME,2)
      CALL PLOTP(AKOUNT,TRUY,ITIME,3)
      WRITE(6,802)
      FORMAT(25X,'YPOSIT VS TIME')
      WRITE(6,900)
      CALL PLOTP(AKOUNT,XDIFF3,ITIME,0)
      WRITE(6,806)
      FORMAT(25X,'DEVIATION FROM TRUE Y')
      WRITE(6,900)
      CALL PLOTP(C3,C1,2,1)
      CALL PLOTP(AKOUNT,X5,ITIME,2)
      CALL PLOTP(AKOUNT,TRUZ,ITIME,3)
      WRITE(6,803)
      FORMAT(25X,'Z DEPTH VS TIME')
      WRITE(6,900)
      CALL PLOTP(C4,C5,2,1)
      CALL PLOTP(X1,X3,ITIME,2)
      CALL PLOTP(TRUX,TRUY,ITIME,3)
      WRITE(6,804)
      FORMAT(25X,'GEOGRAPHIC Y VS X')
      WRITE(6,900)
      END

```

801

805

802

806

803

804

APPENDIX C
TRADITIONAL EXTENDED KALMAN FILTER
THESIS

```

COMMON XKKM1(5),PKKM1(5,5),PKK(5,5),XKK(5)
DIMENS ION PHI(5,5),GAMMA(5,3),R(4,4),PHIPKK(5,5),X1(510),X3(510),
PKTEMP(5,4),H(4,5),HT(5,4),AA(20),CTEMP(5,3),X1(510),X3(510),
IKNUM(5,4),G(5,4),ZI(4),GAMMAT(3,5),Q(5,5),CQVW(3,3),
X5(510),KOUNT(510),AKOUNT(510),PVEC(510),P1(510,5,5),
ITRUX(510),TRUX(510),TRUZ(510),ID(3),C1(2),C2(2),
IXP(5,510),C4(2),C5(2),HROW(5),GOTEM1(4,5),GOTEM2(4,4),
GIDENOM(4,4),ZDIFF(4),GZ(5),PTEM1(5,5),PTEM2(5,5),ZIC(4),ZCS(4),
IEI(5,5),C3(2),ZH(4),C6(2),C7(2),L(4),MM(4),ZC(4),ZIC(4),ZCS(4),
IXDIFF1(510),XDIFF3(510),XDIFF5(510),HYDRO(6,12),
XB(4),YB(4),ZB(4)
N=5
M=3
JS=4
I8=1
I0=1,3
READ(5,200)JTIME
FORMAT(13)
READ(5,201)SIGACC,SIGDIV,SIGCC
FORMAT(3F4,2)
201 SIGCC=SIGCC*3.14159/180.
DO 799 I=1,6
799 READ(7,739){HYDRO(I,J),J=1,12}
739 FORMAT(12F6.1)

C FORM AN IDENTITY MATRIX EI
C
DO 140 I=1,N
DO 140 J=1,N
140 EI(I,J)=0.0
DO 141 I=1,N
141 EI(I,I)=1.0
C LOAD X(0/-1)
CALL VREAD(XKKM1,N)
C LOAD P(0/-1)
CALL MREAD(PKKM1,N,N)
C READ IN CONSTANT MATRICES-GET TRANSPOSES
CALL MREAD(PHI,N,N)
CALL MREAD(R,JS,JS)
CALL MREAD(CQVW,M,M)
CALL MREAD(GAMMA,N,M)

```

```

THE00010
THE00020
THE00030
THE00040
THE00050
THE00060
THE00070
THE00080
THE00090
THE00100
THE00110
THE00120
THE00130
THE00140
THE00150
THE00160
THE00170
THE00180
THE00190
THE00200
THE00210
THE00220
THE00230
THE00240

THE00270
THE00280
THE00290
THE00300
THE00310
THE00320
THE00330
THE00340
THE00350

THE00370
THE00380
THE00390

THE00410
THE00420
THE00430

THE00450
THE00460
THE00470
THE00480
THE00490

```

THE00500
THE00510
THE00620
THE00720
THE00730
THE00740
THE00750
THE00760
THE00770
THE00780
THE00790
THE00800
THE00810
THE00820
THE00830
THE00840
THE00850
THE00860
THE00870
THE00880

THE00990
THE01000
THE01010
THE01020
THE01030
THE01040
THE01050
THE01060
THE01070
THE01080
THE01090
THE01100
THE01110
THE01120
THE01140
THE01190
THE01200
THE01210
THE01220
THE01230
THE01240

THE01260
THE01270
THE01280
THE01290
THE01300

86

THESIS (con't)

```

121 H(KY,J)=HROW(J)
120 CONTINUE
C CALCULATE THE GAIN MATRIX
      CALL PROD(PKKM1,HT,N,N,JS,GNUM)
      CALL PROD(H,PKKM1,JS,N,N,GOTEM1)
      CALL PROD(GOTEM1,HT,JS,N,JS,GDTEM2)
      CALL ADD(GDTEM2,R,JS,JS,GDENOM)
      II=0
      DO 23 IA=1,JS
        DO 23 JA=1,IA
          II=II+1
          AA(II)=GDENOM(JA,IA)
          CALL SINVA(AA,JS,EPS,IER)
          II=0
        DO 24 IA=1,JS
          DO 24 JA=1,IA
            II=II+1
            GDENOM(JA,IA)=AA(II)
            GDENOM(IA,JA)=AA(II)
          CALL PROD(GNUM,GDENOM,N,JS,JS,G)
        C23
      C
      C
      C
      C
      C
      C
      C
      C24
C CALCULATE THE COVARIANCE OF ESTIMATION ERROR P(K/K)

      CALL PROD(G,H,N,JS,N,PTEM1)
      DO 127 ID=1,N
        DO 127 JD=1,N
          PTEM1(ID,JD)=1.*PTEM1(ID,JD)
          CALL ADD(EI,PTEM1,N,N,PTEM2)
          CALL PROD(PTEM2,PKKM1,N,N,N,PKK)
        127
C GET AN ESTIMATE OF THE MEASUREMENT
C CALCULATE THE RESIDUAL
      DO 122 IB=1,JS
        CALL CZHAT3(IB,ZHAT,XB,YB,ZB)
        ZH(IB)=ZHAT
        ZDIFF(IB)=ZI(IB)-ZHAT
      122
C SET THE GATE
      PK1=ABS(PKK(1,1))
      PK3=ABS(PKK(3,3))
      PK5=ABS(PKK(5,5))
      IF((PK1.GE.PK3).AND.(PK1.GE.PK5))P=PK1
      IF((PK3.GE.PK1).AND.(PK3.GE.PK5))P=PK3
      IF((PK5.GE.PK1).AND.(PK5.GE.PK3))P=PK5

```

```

THE01310
THE01320
THE01330

THE01350
THE01360
THE01370
THE01380
THE01390
THE01400
THE01470
THE01480
THE01490
THE01500
THE01510
THE01540
THE01550
THE01560
THE01570
THE01580
THE01600
THE01670
THE01660
THE01590

THE01780
THE01790
THE01830
THE01840
THE01850
THE01860
THE01930
THE01940

THE01970
THE01980
THE01990
THE02000
THE02010
THE02020

THE02040
THE02050
THE02060
THE02070
THE02080
THE02090

```

THESIS (con't)

```

CC      DO 500 LG=1,N
        PGATE=P/4860.
        PGATE=ABS(R(LG,LG))
        GATE=3.*(SORT(PGATE+RGATE))
        IF(ZDIFF(LG).LT.GATE)GO TO 500
        WRITE(6,501)LG,KK
        FORMAT('0',GATE HAS BEEN EXCEEDED STATE VAR',11,'TIME',12)
501      DO 502 LH=1,N
        DO 502 LM=1,JS
502      G(LH,LM)=0.0
        GO TO 503
500      CONTINUE
CC      C CALCULATE THE ESTIMATE
C      503      CALL MMULT(G,ZDIFF,N,JS,GZ)
        DO 124 IC=1,N
124      XKK(IC)=XKKM1(IC)+GZ(IC)
        XDIFFF1(KK)=TRUX(KK)-XKK(1)
        XDIFFF3(KK)=TRUY(KK)-XKK(3)
        XDIFFF5(KK)=TRUZ(KK)-XKK(5)
C      C
C      C CALCULATE THE PREDICTIONS FOR PKKM1
        CALL QFIND(KK,SIGACC,SIGDIV,SIGCC,A14,Q)
        CALL PROD(PHI,PKK,N,N,N,PHIPKK)
        CALL PROD(PHIPKK,PHIT,N,N,N,PKTEMP)
        CALL ADD(PKTEMP,Q,N,N,PKKM1)
C      C CALCULATE THE PREDICTIONS FOR XKKM1
        CALL MMULT(PHI,XKK,N,N,XKKM1)
        DO 41 IG=1,N
41      XP(IG,KK)=XKK(IG)
        DO 38 IZ=1,N
        DO 39 JJ=1,N
        P1(KK,IZ,JJ)=PKK(IZ,JJ)
        CONTINUE
        C1(1)=290.
        C1(2)=310.
        C3(1)=0.
        C3(2)=510.
        C4(1)=0.
        C4(2)=45000.
        C5(1)=0.

```

```

THE02100
THE02110
THE02120
THE02130
THE02140
THE02150
THE02160
THE02170
THE02180
THE02190
THE02200
THE02210
THE02220
THE02230
THE02250
THE02260
THE02270
THE02280
THE02290
THE02300
THE02310
THE02330
THE02340
THE02620
THE02640
THE02710
THE02720
THE02730
THE02740
THE02760
THE02770
THE02780
THE02790
THE02800
THE02810
THE02820
THE02830
THE02860
THE02870
THE02880
THE02890
THE02900
THE02910
THE02920
THE02930

```

THESIS (con't)

THE02940
THE02950
THE02960
THE02970
THE02980
THE02990
THE03000
THE03010
THE03020
THE03030
THE03040
THE03050
THE03060
THE03070
THE03080
THE03090
THE03100
THE03110
THE03120
THE03130
THE03140
THE03150
THE03160
THE03170
THE03180
THE03190
THE03200
THE03210
THE03220
THE03230
THE03240
THE03250
THE03260
THE03270
THE03280
THE03290
THE03300
THE03310
THE03320
THE03330
THE03340
THE03350
THE03360
THE03370
THE03380
THE03390
THE03400
THE03410

```

C5(2)=10000.
DO 91 KK=1, ITIME
X1(KK)=XP(1, KK)
X3(KK)=XP(3, KK)
X5(KK)=XP(5, KK)
DO 92 IK=1, ITIME
KOUNT(IK)=IK
DO 113 IF=1, ITIME
AKOUNT(IF)=FLOAT(KOUNT(IF))
DO 238 IZ=1, N
JJ=IZ
DO 240 KK=1, ITIME
PVEC(KK)=PI(KK, IZ, JJ)
WRITE(6, 900)
CALL PLOTP(AKOUNT, PVEC, ITIME, 0)
238 WRITE(6, 800) IZ, JJ
800 FORMAT(23X, 'PKK(', I1, ', ', I1, ', ', 'VS TIME')
900 WRITE(6, 900)
CALL PLOTP(C3, C4, 2, 1)
CALL PLOTP(AKOUNT, X1, ITIME, 2)
CALL PLOTP(AKOUNT, TRUX, ITIME, 3)
801 WRITE(6, 801) 'XPOSIT VS TIME'
FORMAT(25X, 'XPOSIT VS TIME')
WRITE(6, 900)
CALL PLOTP(AKOUNT, XDIFF1, ITIME, 0)
805 WRITE(6, 805) 'DEVIATION FROM TRUE X'
FORMAT(25X, 'DEVIATION FROM TRUE X')
WRITE(6, 900)
CALL PLOTP(C3, C5, 2, 1)
CALL PLOTP(AKOUNT, X3, ITIME, 2)
CALL PLOTP(AKOUNT, TRUY, ITIME, 3)
802 WRITE(6, 802) 'YPOSIT VS TIME'
FORMAT(25X, 'YPOSIT VS TIME')
WRITE(6, 900)
CALL PLOTP(AKOUNT, XDIFF3, ITIME, 0)
806 WRITE(6, 806) 'DEVIATION FROM TRUE Y'
FORMAT(25X, 'DEVIATION FROM TRUE Y')
WRITE(6, 900)
CALL PLOTP(C3, C1, 2, 1)
CALL PLOTP(AKOUNT, X5, ITIME, 2)
CALL PLOTP(AKOUNT, TRUZ, ITIME, 3)
803 WRITE(6, 803) 'Z DEPTH VS TIME'
FORMAT(25X, 'Z DEPTH VS TIME')
WRITE(6, 900)
CALL PLOTP(C4, C5, 2, 1)
CALL PLOTP(X1, X3, ITIME, 2)
CALL PLOTP(TRUX, TRUY, ITIME, 3)

```

THESIS (con't)

THE03420
THE03430
THE03440
THE03450

WRITE(6,804)
FORMAT(25X,'GEOGRAPHIC Y VS X')
WRITE(6,900)
END

804

APPENDIX D - UTILITY SUBROUTINES SUBROUTINE AUX

AUX00010
AUX00020
AUX00030
AUX00040
AUX00050
AUX00060
AUX00070
AUX00080
AUX00090
AUX00100
AUX00110
AUX00120
AUX00130
AUX00140
AUX00150
AUX00160
AUX00170
AUX00180
AUX00190
AUX00200
AUX00210
AUX00220
AUX00230
AUX00240
AUX00250
AUX00260
AUX00270
AUX00280
AUX00290
AUX00300
AUX00310
AUX00320
AUX00330
AUX00340
AUX00350
AUX00360
AUX00370
AUX00380
AUX00390
AUX00400
AUX00410
AUX00420
AUX00430
AUX00440
AUX00450
AUX00460
AUX00470
AUX00480

1 SUBROUTINE PROD(A,B,N,M,L,C)
DIMENSION A(N,M),B(M,L),C(N,L)
DO 1 I=1,N
DO 1 J=1,L
C(I,J)=0.
DO 2 I=1,N
DO 2 J=1,L
DO 2 K=1,M
C(I,J)=C(I,J)+A(I,K)*B(K,J)
RETURN
END
2 SUBROUTINE MMULT(A,B,N,C)
DIMENSION A(N,N),B(N),C(N)
DO 3 I=1,N
C(I)=0.
DO 4 J=1,N
C(I)=C(I)+A(I,J)*B(J)
CONTINUE
RETURN
END
3 SUBROUTINE VMJLT(A,B,N,C)
DIMENSION A(N),B(N)
C=0.
DO 6 I=1,N
C=C+A(I)*B(I)
RETURN
END
6 SUBROUTINE MREAD(A,N,M)
DIMENSION A(N,M)
DO 10 I=1,N
READ(3,11)(A(I,J),J=1,M)
FORMAT(5F10.5)
DO 20 I=1,N
WRITE(6,21)(A(I,J),J=1,M)
FORMAT(5F12.8)
RETURN
END
10 SUBROUTINE TRANS(A,N,M,B)
11 DIMENSION A(N,M),B(M,N)
20 DO 13 I=1,N
21 DO 13 J=1,M
B(J,I)=A(I,J)
RETURN
END
13 SUBROUTINE ADD(A,B,N,M,C)
DIMENSION A(N,M),B(N,M),C(N,M)
DO 15 I=1,N
DO 15 J=1,M

AUX (con't)

```

AUX000490
AUX000500
AUX000510
AUX000520
AUX000530
AUX000540
AUX000550
AUX000560
AUX000570
AUX000580
AUX000590
AUX000600
AUX000610
AUX000620
AUX000630
AUX000640
AUX000650
AUX000660
AUX000670
AUX000680
AUX000690
AUX000700

```

```

15  C(I,J)=A(I,J)+B(I,J)
    RETURN
    END
    SUBROUTINE VREAD(A,N)
    DIMENSION A(N)
    READ(4,23) (A(I),I=1,N)
    FORMAT(5F10.5)
    RETURN
    END
    SUBROUTINE DREAD(A,JS)
    DIMENSION A(JS)
    READ(2,33) (A(I),I=1,JS)
    FORMAT(4F10.5)
    RETURN
    END
    SUBROUTINE TRREAD(A,M)
    DIMENSION A(M)
    READ(1,43) (A(I),I=1,M)
    FORMAT(3F10.5)
    RETURN
    END
43

```

```

SUBROUTINE CZHAT (I,ZHAT)
COMMON XKKM1(5)
VEL=4860.
IF(I.EQ.1) ZHAT=1./VEL*((XKKM1(1)+15.)*2)+((XKKM1(3)+15.)*2)
1+((XKKM1(5)+15.)*2)**0.5
IF(I.EQ.2) ZHAT=1./VEL*((XKKM1(1)-15.)*2)+((XKKM1(3)+15.)*2)
1+((XKKM1(5)+15.)*2)**0.5
IF(I.EQ.3) ZHAT=1./VEL*((XKKM1(1)+15.)*2)+((XKKM1(3)-15.)*2)
1+((XKKM1(5)+15.)*2)**0.5
IF(I.EQ.4) ZHAT=1./VEL*((XKKM1(1)+15.)*2)+((XKKM1(3)+15.)*2)
1+((XKKM1(5)-15.)*2)**0.5
RETURN
END

```

ZHA00010
 ZHA00020
 ZHA00030
 ZHA00040
 ZHA00050
 ZHA00060
 ZHA00070
 ZHA00080
 ZHA00090
 ZHA00100
 ZHA00110
 ZHA00120
 ZHA00130

CZHAT - USED IN SINGLE ARRAY TRACKING

```

SUBROUTINE CZHAT3 (I,ZHAT,XB,YB,ZB)
COMMON XKKM1(5)
DIMENSION XB(4),YB(4),ZB(4)
VEL=4860.
XO=XB(1)
YO=YB(1)
ZO=ZB(1)
ZHAT=1./VEL*((XKKM1(1)-XO)*2)+((XKKM1(3)-YO)*2)+((XKKM1(5)
1-ZO)*2)**0.5
RETURN
END

```

CZH00010
 CZH00020
 CZH00030
 CZH00040
 CZH00050
 CZH00060
 CZH00070
 CZH00080
 CZH00090
 CZH00100
 CZH00110

CZHAT3 - USED IN MULTIPLE ARRAY TRACKING

SUBROUTINE CHROW

CHR00010
CHR00020
CHR00030
CHR00040
CHR00050
CHR00060
CHR00070
CHR00080
CHR00090
CHR00100
CHR00110
CHR00120
CHR00130
CHR00140
CHR00150
CHR00160
CHR00170
CHR00180
CHR00190
CHR00200
CHR00210
CHR00220
CHR00230
CHR00240
CHR00250
CHR00260
CHR00270
CHR00280
CHR00290
CHR00300
CHR00310
CHR00320

```

SUBROUTINE CHROW(I,HROW)
COMMON XKKM1(5)
DIMENSION HROW(5)
VEL=4860.
DENOM1=((XKKM1(1))+15.)*2)+((XKKM1(3))+15.)*2)+((XKKM1
1(5))+15.)*2))*0.5
DENOM2=((XKKM1(1))-15.)*2)+((XKKM1(3))+15.)*2)+((XKKM1
1(5))+15.)*2))*0.5
DENOM3=((XKKM1(1))+15.)*2)+((XKKM1(3))-15.)*2)+((XKKM1
1(5))+15.)*2))*0.5
DENOM4=((XKKM1(1))+15.)*2)+((XKKM1(3))+15.)*2)+((XKKM1
1(5))-15.)*2))*0.5
A1=1.
A2=1.
A3=1.
DENOM=DENOM1
IF(I.EQ.2)DENOM=DENOM2
IF(I.EQ.3)DENOM=DENOM3
IF(I.EQ.4)DENOM=DENOM4
IF(I.EQ.2)A1=-1.
H1=(1./VEL)*((XKKM1(1))+A1*15.)/DENOM)
IF(I.EQ.3)A2=-1.
H2=(1./VEL)*((XKKM1(3))+A2*15.)/DENOM)
IF(I.EQ.4)A3=-1.
H3=(1./VEL)*((XKKM1(5))+A3*15.)/DENOM)
HROW(1)=H1
HROW(3)=H3
HROW(5)=H5
DO 27 J=2,4,2
HROW(J)=0.
27 RETURN
END

```

CHROW3

CHR00010
CHR00020
CHR00030
CHR00040
CHR00050
CHR00060
CHR00070
CHR00080
CHR00090
CHR00100
CHR00110
CHR00120
CHR00130
CHR00140
CHR00150
CHR00160

```

SUBROUTINE CHROW3 (I,HROW,XB,YB,ZB)
COMMON XKKM1(5)
DIMENSION HROW(5),XB(4),YB(4),ZB(4)
VEL=4860.
X0=XB(1)
Y0=YB(1)
Z0=ZB(1)
DENOM=((XKKM1(1)-X0)**2)+((XKKM1(2)-Y0)**2)+((XKKM1(5)-Z0)
1 *2)**2)**0.5
HROW(1)=(1./VEL)*((XKKM1(1)-X0)/DENOM)
HROW(3)=(1./VEL)*((XKKM1(3)-Y0)/DENOM)
HROW(5)=(1./VEL)*((XKKM1(5)-Z0)/DENOM)
DO 27 J=2,4,2
HROW(J)=0.
RETURN
END

```

27

CHROW3 - USED IN MULTIPLE ARRAY TRACKING

AD-A064 896

NAVAL POSTGRADUATE SCHOOL MONTEREY CALIF
REAL TIME KALMAN FILTERING FOR TORPEDO RANGE TRACKING.(U)
DEC 78 D M DWYER

F/6 17/1

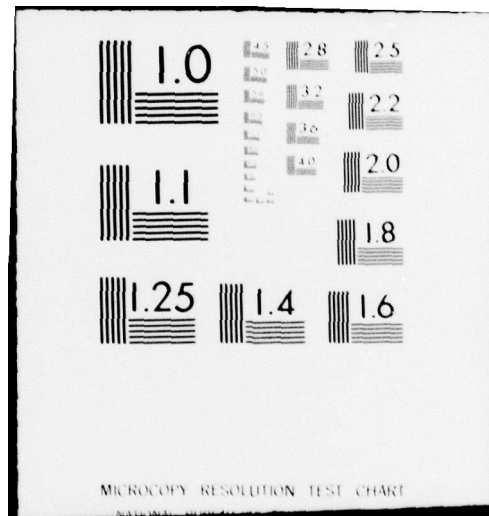
UNCLASSIFIED

NL

2 OF 2
AD
A 064 896



END
DATE
FILMED
4-79
DDC



QFIND

QF I 00010
QF I 00020
QF I 00030
QF I 00040
QF I 00050
QF I 00060
QF I 00070
QF I 00080
QF I 00090
QF I 00100
QF I 00110
QF I 00120
QF I 00130
QF I 00140
QF I 00150
QF I 00160
QF I 00170
QF I 00180
QF I 00190
QF I 00200
QF I 00210
QF I 00220
QF I 00230
QF I 00240
QF I 00250
QF I 00260
QF I 00270
QF I 00280
QF I 00290
QF I 00300
QF I 00310
QF I 00320
QF I 00330
QF I 00340
QF I 00350
QF I 00360
QF I 00370

```

SUBROUTINE QFIND(K,SIGACC,SIGDIV,SIGCC,A,Q)
COMMON XHKK(5),PKK(5,5),PKK(5,5),XHKK(5)
DIMENSION Q(5,5)
IF(K.NE.1) GO TO 15
DO 10 I=1,5
  DO 10 J=1,5
    Q(I,J)=0
    SIGACC=SIGACC**2
    Q(5,5)=(SIGDIV*A)**2
    SIGCC=SIGCC**2
    G1=(A**2)/2.0
    G2=G1**2
    G3=A*G1
    A2=A**2
    A1=XHKK(2)**2+XHKK(4)**2
    A3=XHKK(2)/SQRT(A1)
    B=XHKK(4)
    C=XHKK(4)/SQRT(A1)
    D=XHKK(2)
    E1=(A3**2)*SIGACC+(B**2)*SIGCC
    E12=(A3**2)*SIGACC-B*D*SIGCC
    E2=(C**2)*SIGACC+(D**2)*SIGCC
    Q(1,1)=E1*G2
    Q(1,2)=E12*G2
    Q(1,3)=G3*E12
    Q(1,4)=G3*E12
    Q(2,3)=G3*E12
    Q(2,4)=G3*E12
    Q(3,3)=G3*E12
    Q(3,4)=G3*E12
    Q(4,4)=A2*E2
    DO 27 I=1,4
      DO 27 J=1,4
        Q(I,J)=Q(I,J)
      RETURN
    END
  10
  15
  27

```

LIST OF REFERENCES

1. NAVAL TORPEDO STATION REPORT 1382, NAVTORPSTA 3-D RANGE COMPUTER SOFTWARE CONVERSION PLAN, J. J. Hirschfelder, November 1977.
2. NAVAL TORPEDO STATION REPORT 1030, Principle of the Three dimensional Range, Revision 1, Enclosures (1) and (2).
3. Benson, Eric J., An Application of Kalman Filtering To Underwater Tracking, Masters Thesis, Naval Postgraduate School, December 1976.
4. Mitschang, George W., An Application of Nonlinear Filtering Theory to Passive Target Location and Tracking, PhD Thesis, Naval Postgraduate School, June 1974.
5. Kirk, Donald E., EE 4413 Optimal Control Systems Class notes (unpublished), Naval Postgraduate School, 1975.

INITIAL DISTRIBUTION LIST

| | No. Copies |
|--|------------|
| 1. Defense Documentation Center Cameron Station Alexandria, Virginia 22314 | 2 |
| 2. Library, Code 0142 Naval Postgraduate School Monterey, California 93940 | 2 |
| 3. Department Chairman, Code 62 Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940 | 1 |
| 4. Professor H. A. Titus, Code 62Ts Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940 | 5 |
| 5. Professor G. L. Sackman, Code 62Sa Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940 | 1 |
| 6. Commanding Officer Naval Underwater Weapons Engineering Station Kayport, Washington 98345 | 5 |
| 7. Lieutenant Dennis M. Dwyer 4953 Klein Court Woodbridge, Virginia 22193 | 2 |